HAL
open science

Digitization, compression and reconstruction of a large-scale radio traffic for the Internet of Things

Esteban Selva

To cite this version:

Esteban Selva. Digitization, compression and reconstruction of a large-scale radio traffic for the Internet of Things. Signal and Image processing. CentraleSupélec, 2021. English. NNT : 2021CSUP0005 . tel-03447017

HAL Id: tel-03447017

https://theses.hal.science/tel-03447017v1

Submitted on 24 Nov 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE DE DOCTORAT DE

CENTRALESUPELEC

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : Télécommunications

Par
Esteban SELVA

Numérisation, compression et reconstruction d'un trafic radio large échelle pour l'internet des objets

Thèse présentée et soutenue à Cesson-Sévigné, le 10 juin 2021
Unité de recherche : UMR 6164 – IETR (équipe SCEE)
Thèse N° : 2021CSUP0005

Rapporteurs

Damien Roque	Professeur, ISAE-SUPAERO
Claire Goursaud	Maître de conférences, INSA Lyon

Composition du Jury

Président :	Benoît Geller	Professeur, ENSTA Paris
Examinateurs :	Laurent Ros	Professeur, Grenoble INP
	Benoît Geller	Professeur, ENSTA Paris
Dir. de thèse :	Yves Louët	Professeur, CentraleSupelec
Co-dir. de thèse :	Apostolos Kountouris	Ingénieur de recherche, Orange Labs Grenoble

Invité(s)

Jean Schwoerer	Ingénieur de recherche, Orange Labs Grenoble
Christophe Moy	Professeur, Université de Rennes 1

Attention.
Cette tsèhe n'est pas une tsèhe sur le cyclimse.
Merci de votre compréhension.

Remerciements

Ce manuscrit est le fruit d'une thèse CIFRE menée durant un peu plus de trois ans dans le cadre d'une convention entre l'équipe de recherche Signal, Communication et Électronique embarquée (SCEE), membre de l'Institut l'Électronique et de Télécommunications numéRiques (IETR), localisée à CentraleSupelec à Rennes, et de l'équipe Orange Innovation/IT-S/BIZZ/MIS/CITY d'Orange, située à Orange Labs Grenoble. Ayant travaillé dans deux équipes, attendez-vous à une section de remerciements particulièrement fournie.

Je voudrais en tout premier lieu remercier chaleureusement Yves LOUET, mon directeur de thèse, et Apostolos KOUNTOURIS, mon encadrant chez Orange, pour leur implication dans mon travail. Merci à toi, Yves, d'avoir fait l'effort de nous rendre fréquemment visite à Grenoble pour nos réunions constructives et de qualité. Apostolos, merci d'avoir toujours posé les questions qui font avancer.

Je tiens ensuite à remercier les rapporteurs de ce travail de thèse, Claire GOURSAUD et Damien ROQUE, pour leur implication dans la relecture de mon manuscrit et pour la qualité de leurs rapports. Je remercie également Benoît GELLER d'avoir présidé ce jury et de m'avoir fait découvrir le monde des réseaux de communications au cours de mes études à l'ENSTA Paris. Je souhaite remercier Laurent ROS pour son rôle d'examinateur au sein du jury, et Jean SCHWOERER et Christophe MOY pour leur présence dans ce jury en tant que membres invités. À tous les membres du jury, j'adresse des remerciements pour les discussions intéressantes que nous avons pu avoir au cours de la soutenance.

Je souhaite à présent remercier Vincent SAVAUX et Moïse DJOKO-KOUAM pour les conseils prodigués dans le cadre du comité de suivi individuel du doctorant.

Travailler à cheval entre deux laboratoires situés respectivement à Rennes et à Grenoble représente une chance mais également des contraintes logistiques. Habitant à Grenoble, je me rendais (avant la pandémie) à Rennes une semaine tous les deux ou trois mois environ. Je garderai un excellent souvenir de mes séjours rennais, et ceci grâce à mes collègues de CentraleSupélec: merci à Lilian, Bastien, Adrien, Corentin, Morgane, Éloïse, Simon, Nabil, Rémi, Rami et Christophe. Malgré ma faible durée effective à Rennes, vous m'avez intégré dignement et je me souviendrai (en bien) de l'accueil breton.

C'est à Grenoble que j'ai passé la majeure partie de ma thèse : j'en garderai de magnifiques souvenirs. Comment ne pas remercier Quentin, mon collègue, ancien maître de stage et désormais ami, pour les sorties en trail, l'hébergement, l'initiation au gravel bike et tant d'autres ? Un grand merci à mes collègues du bureau E 988, Wissal et Dihia, pour les discussions enflammées et pour le bon esprit au cours de nos trois thèses quasi simultanées. Merci

aux autres collègues d'Orange Labs pour les discussions et les bons moments, notamment Louis-Adrien, Guillaume, Marion, Nadège, Pauline, Philippe et Simon. Mes pensées vont également vers Céline, partie bien trop tôt. Et bien sûr, je remercie vivement les forces vives de la nation Paul et Maël: nous sommes enfin les *Young Doctors*!

Je souhaite remercier quelques autres illustres Grenoblois: Rémi et Estelle pour les belles sorties sportives et en montagne ainsi que pour les repas simples mais délicieux, et JM pour les soirées et les sorties à vélo. Merci aussi à mes amis d'ailleurs (principalement de l'ENSTA) pour leur soutien et leur amitié au cours de la thèse: Manu, Jordan, Nico, Julie et Adrien, Mosssieur Gontran, FS, Pifou, Robin et Juliette.

J'ai eu la chance d'être embauché chez Orange au terme de mon contrat de thèse, et j'ai rédigé une part importante de ce manuscrit alors que j'avais rejoint une nouvelle équipe. Ainsi, j'aimerais remercier d'une part Orange de m'avoir fourni les conditions matérielles me permettant d'achever ma thèse sereinement, et d'autre part mon nouveau chef Azeddine pour sa patience et sa compréhension au cours des quelques mois de transition entre la thèse et le nouveau poste. Je remercie également vivement mes nouveaux collègues de m'avoir intégré dans l'équipe alors que je finissais ma thèse.

J'ai le bonheur d'avoir été en permanence accompagné et soutenu par ma famille aimante au cours de ces travaux de thèse. Merci infiniment à mon frère Virgile, à mes parents Isabelle et Thierry, à mes grand-parents Christiane et Bernard, à Geneviève et Hubert, à Liliane, à Mimi, à la famille David-Kébala et à Gilles. Et bien sûr un immense merci à Camille, ma dulcinée. Merci à vous d'être dans ma vie.

Pour terminer je tiens à remercier ma bonne étoile, sans laquelle je n'en serais pas où je suis aujourd'hui.

Merci à tous ceux qui ont contribué à l'écriture de ce manuscrit.

Contents

Remerciements **5**

Table of contents **8**

List of figures **11**

List of tables **15**

List of acronyms **17**

List of notations **20**

List of publications of the author during the PhD **21**

Résumé en français **23**

Introduction 23
État de l'art 24
Outils pour la radio cognitive 25
Paramétrer l'ÉMT 27
Une infrastructure d'échantillonnage tournée vers l'IdO fondée sur l'ÉMT 28
Conclusion et perspectives 30

General Introduction **31**

0.1 History and context of wireless transmission-collecting infrastructure 31
0.1.1 Centralized versus decentralized signal processing 31
0.1.2 The Internet of Things: reshuffling the cards of service requirements . . . 33
0.1.3 Conclusion 34
0.2 Stakes of the present study 35
0.2.1 Preliminary considerations: scope of the study 35
0.2.2 Efficient sampling and how we can do it 35
0.2.3 Prototype of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure: use case of this PhD thesis 39
0.3 Plan of the thesis and contributions 41

1 Review of Compressed Sensing and Spectrum Sensing **44**
1.1 Introduction . . . 44
1.2 Theoretical aspects of Compressed Sensing . . . 45
1.2.1 Preliminary considerations on signal and vector spaces . . . 45
1.2.2 Signal sparsity . . . 47
1.2.3 Sampling matrix properties . . . 49
1.2.4 Sampling matrix construction . . . 52
1.2.5 Beyond properties of guarantee of recovery . . . 54
1.2.6 Signal recovery . . . 56
1.2.7 Conclusion on the theoretical aspects of CS . . . 60
1.3 Implementation of Compressed Sensing in Telecommunications . . . 60
1.3.1 Applications of Compressed Sensing in Telecommunications . . . 61
1.3.2 Compressed Sensing Radiofrequency Architectures . . . 64
1.3.3 Conclusion on the Applications and Architectures of CS . . . 73
1.4 Spectrum Sensing . . . 74
1.4.1 Introduction: Dynamic Spectrum Access and Sampling Infrastructure . . . 74
1.4.2 Narrowband Spectrum Sensing . . . 77
1.4.3 Wideband Spectrum Sensing . . . 82
1.4.4 Conclusion on Spectrum Sensing . . . 84
1.5 Conclusion . . . 84

2 Spectrum Sensing: Noise Variance and Support Estimations of Sparse Signals **86**
2.1 Introduction . . . 86
2.1.1 Motivation for estimating the noise variance and the signal support . . . 86
2.1.2 Chapter Outline . . . 87
2.2 Noise Variance Estimation . . . 88
2.2.1 Introduction to Noise Variance Estimation . . . 88
2.2.2 System Model . . . 89
2.2.3 Noise Variance Estimation based on K-means clustering . . . 90
2.2.4 Simulation Results and Discussions . . . 96
2.2.5 Conclusion and Perspectives on Noise Variance Estimation . . . 101
2.3 Support Estimation . . . 101
2.3.1 Introduction to Support Estimation . . . 101
2.3.2 System Model . . . 102
2.3.3 Sample-wise energy-based signal detection . . . 103
2.3.4 Enhancing sample-wise energy detection with the moving average . . . 106
2.3.5 Simulations: support estimation . . . 109
2.3.6 Application to a telecommunications signal: the filtered QPSK . . . 112
2.3.7 Conclusion on Support Estimation . . . 115
2.4 Conclusion . . . 116

3 Configuring the MRS settings **117**
3.1 Introduction . . . 117
3.2 Link between Sampling Rates and the Measurement Matrix Rank . . . 118
3.2.1 Recovery guarantee properties do not characterize the typical behavior of a sampling setup . . . 118
3.2.2 Motivation for using the matrix rank as an indicator of the quality of recovery . . . 119
3.2.3 On the information acquisition process underlying the MRS . . . 119
3.2.4 Sampling rates and matrix rank . . . 120
3.2.5 Illustration of the link between sampling rates and sampling matrix rank 121
3.3 Empirical relations between MRS settings and recovery performance . . . 122
3.3.1 Introduction . . . 122
3.3.2 Simulation Results: Recovering k-sparse noiseless signals . . . 123
3.3.3 Application: Recovering noisy QPSK-based signals . . . 128
3.4 Conclusion and Perspectives . . . 131

4 Experimental analysis of an MRS-based sampling infrastructure for IoT traffic **134**
4.1 Introduction . . . 134
4.2 Presentation of the sampling infrastructure prototype . . . 135
4.3 The IoT-oriented traffic generator . . . 137
4.3.1 Parameters of the traffic generator . . . 138
4.3.2 Modulation types . . . 139
4.3.3 Traffic types . . . 141
4.4 Validation of the toolkit introduced in Chapter 2 . . . 143
4.4.1 Introduction . . . 143
4.4.2 Practical considerations when using our toolkit . . . 144
4.4.3 Using the K-means-based noise variance estimator in our sampling infrastructure . . . 145
4.4.4 Signal support estimation in each MRS branch . . . 147
4.4.5 Combination of partial support estimates . . . 155
4.4.6 Conclusion . . . 158
4.5 Impact of the reduction procedure and solver . . . 160
4.5.1 Introduction . . . 160
4.5.2 Presentation of tested solvers . . . 161
4.5.3 Simulation results . . . 163
4.5.4 Low SNR regime . . . 172
4.5.5 Discussion on the performance of solvers . . . 174
4.5.6 Conclusion on the analysis of solvers . . . 174
4.6 Towards infrastructure adaptivity . . . 175
4.6.1 Characteristics impacting the performance of the MRS-based infrastructure175

4.6.2 Adapting to radio environmental changes 176
4.6.3 Conclusion on infrastructure adaptivity 178
4.7 Summary of our experimental findings 178

Conclusion and perspectives **180**
Summary of our contributions and perspectives 180
Extension: towards a multi-site sampling infrastructure 183

A Appendix relative to Chapter 2 **185**
A.1 Relationship between P_D, η and $E_{\min}$ in Section 2.3.3.4 185

References **199**

List of Figures

0.1 Description de l'estimateur de la variance du bruit fondé sur les K-moyennes. . . 25
0.2 Estimation de la variance du BBGA en fonction du RSB pour différents taux d'occupation du spectre fréquentiel. 26
0.3 Courbe ROC de l'ES sans fenêtrage pour différentes valeurs d'énergie $E_{\min}$ (unités arbitraires). 27
0.4 Transitions entre phases de succès et d'échec pour un (a) nombre de mesures normalisé; (b) rg A normalisé. QFR-MRS = ÉMT avec M_i premiers entre eux deux-à-deux. DR-MRS = ÉMT avec M_i non deux-à-deux premiers entre eux. . . 28
0.5 Racine de l'erreur quadratique moyenne relative en fonction du nombre de mesures normalisé pour un faible volume de trafic et un RSB élevé (RSB = 3 dB). 29
0.6 Proposed scenario of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure. 39

1.1 Sparsifying transform: transforming a non-sparse signal s into a k-sparse signal x through multiplication by a sparsifying basis Φ. 48
1.2 Compressed Sensing: collecting $m \ll n$ measurements through a sampling matrix A. The m-sized vector y contains all the information present in n-sized x and represents it in fewer coefficients. 49
1.3 Processing CS-acquired signals. 50
1.4 Phase transitions for two different recovery algorithms ((a): ℓ_1 minimization (b): StOMP, see Section 1.2.6). The sampling matrices are Gaussian with $n = 1600$. The different shades represent fractions of cases in which the signal recovery algorithm successfully finds the sparsest solution (darkest: 0% success rate, brightest: 100% success rate). 55
1.5 Description of MCS. 65
1.6 Multi-Coset sampling grid. The crosses represent the sampling instants $kLT+c_iT$ of MCS for $p = 4$ branches with $L = 7$ and $c_i = 0$, 1, 3 and 6 respectively. T is the Nyquist period. 65
1.7 Undersampling a sparse multi-band signal at sampling rates $2f_{s1}$ and $2f_{s2}$ (signals represented in the frequency domain). 67
1.8 Description of MRS. 68
1.9 Multi-Rate sampling grid. This grid displays the Dirac impulses (sampling instants) for $L = 4$ branches with $M_i = 5$, 4, 3 and 6 respectively. 69
1.10 Description of the MWC. 71

1.11 Mixing function $p_i(t)$. 72
1.12 High-level flowchart describing the acquisition-storage part of our proposed infrastructure. The test for presence of signal of interest is carried out using SS techniques. 76
1.13 Classification of spectrum sensing techniques, as proposed by the authors of [1]. . 77
1.14 Power spectrum densities (PSD) of a (a) NB; (b) WB signal (semilog scale). . . . 78

2.1 Description of our K-means-based method for blind noise variance estimation. . . 91
2.2 Energy of noiseless signal X, its support (black intervals), and noisy signal $Y = X + W$. Signals are represented in the linear scale. 92
2.3 Before and after the preprocessing step. Signals are represented in the log scale for better visibility. The window length is $w_l = 30$ samples. 92
2.4 Average silhouette widths for different number of clusters. 93
2.5 Histogram of the values in the MMC before postprocessing, based on our example signal. The values that remain after the postprocessing step are located on the left-hand side of the vertical line. 95
2.6 Noise variance estimation $\hat{\sigma}_w^2$ vs SNR (dB) for various occupancy ratios ρ. $N = 5000$ samples, $w_l = 30$ samples. 98
2.7 Noise variance estimation $\hat{\sigma}_w^2$ vs SNR (dB) for various smoothing window lengths w_l. $N = 5000$ samples, $\rho = 70\%$. 99
2.8 Maximum relative smoothing window size vs relative channel width, subject to estimation performance constraints. 100
2.9 $\hat{\sigma}_w^2$ vs SNR for various occupancy ratios. The SoI X is a sum of filtered QPSKs. 100
2.10 Required SNR (dB) versus window length w_l, for various (P_D, P_{FA}) pairs. The occupancy ratio is $\rho = 40\%$. 108
2.11 ROC curves of empirical sample-wise energy detection for varying values of $E_{\min}$. 110
2.12 ROC curves of sample-wise energy detection with moving average and varying window lengths w_l. 111
2.13 ROC curves of sample-wise energy detection with moving average for varying fractions of NH samples ν. 111
2.14 ROC curves of our support estimator with varying window lengths w_l, applied to (i) QPSK-based SoIs (solid lines); (ii) SoIs based on the constant energy model (dashed lines) 113
2.15 Fraction e_d of energy present in the support estimate $\hat{K}$ vs proportion of false alarms p_{fa} with varying window lengths w_l, applied to (i) QPSK-based SoIs (solid lines); (ii) constant energy SoIs (dashed lines). 115

3.1 Multi-Rate sampling grid. This grid displays the Dirac impulses (sampling instants) for $L = 4$ branches with $M_i = 5$, 4, 3 and 6 respectively. During the duration Δ of the time window, only the samples comprised within the dashed area provide new information. Samples outside of the dashed area are redundant. 121

3.2 Phases for (a) QFR-MRS; (b) DR-MRS. Black phase: signal recovery is unsuccessful. White phase: signal recovery is successful. The solid line displays the transition between the two phases. 124
3.3 Phase transitions vs normalized (a) number of measurements; (b) matrix rank. . 125
3.4 Phase transitions in the $\rho - \delta$ plane for varying number of branches L, using the CVXPY ℓ_1-minimization solver. 126
3.5 Phase transitions in the $\rho - \delta$ plane for varying number of branches L, using a Least-Squares-based solver. 128
3.6 BER vs. (a) E_b/N_0; (b) MRS matrix rank. Matrix rank maximum value is 1024. 130
3.7 E_b/N_0 degradation vs matrix rank reduction for original $E_b/N_0 = 6$ dB. 131
3.8 BER vs. MRS matrix rank with varying number of branches L for $E_b/N_0 =$ (a) 2 dB ; (b) 6 dB. Matrix rank maximum value is 1024. 132

4.1 Schematic describing our proposed MRS-based sampling infrastructure. 136
4.2 2D time-frequency traffic visualizations for average occupancy ratios of (a) 2.5%; (b) 10%; (c) 25%. 142
4.3 Optimal undersampling ratios δ_i for successful noise variance estimation with the KNVE versus SNR, for different sampling durations (represented by the number of 20-µs blocks) and different traffic scenarios. 146
4.4 Probability P_{FA} of type II errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i for various SNR regimes and traffic scenarios. The lower the P_{FA}, the better. . 151
4.5 Probability P_{FA} of type I errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i for various SE window lengths w_l, in different SNR regimes and traffic scenarios. The lower the P_{FA}, the better. 153
4.6 Probability P_{FA} of type I errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i in traffic scenario 1 for various acquisition durations in the (a) high SNR regime; (b) low SNR regime. The lower the P_{FA}, the better. 154
4.7 Illustration of the support expansion procedure. The estimates are centered around 0 for visual clarity. The M_i-sized partial support estimate (solid line) is periodically reproduced to form the N-sized expanded support estimate (dotted line). 156
4.8 P_{FA} versus undersampling ratio δ for support combination of L perfect partial supports ($P_D = 1$, $P_{FA} = 0$), each computed in an MRS branch. 157
4.9 P_{FA} versus undersampling ratio δ for support combination of $L = 4$ partial support estimates, each computed in an MRS branch, for various SNR regimes. The dashed line represents the performance in optimal conditions. 159
4.10 Relative RMSE versus undersampling ratio δ for traffic scenario 1 (low traffic) in the high SNR regime (SNR = 3 dB). 163
4.11 Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). 165

4.12 Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). Solvers use the actual signal support for USLE resolution (when applicable). 166
4.13 Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). 167
4.14 Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). Solvers use the actual signal support for USLE resolution (when applicable). 168
4.15 Relative RMSE versus undersampling ratio δ for traffic scenario 1 (low traffic) in the medium SNR regime (SNR = -3 dB). 169
4.16 Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the medium SNR regime (SNR = 3 dB). 170
4.17 Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the medium SNR regime (SNR = 3 dB). Solvers use the actual signal support for USLE resolution (when applicable). 171
4.18 Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the medium SNR regime (SNR = 6 dB). 171
4.19 Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the medium SNR regime (SNR = 3 dB). Solvers use the actual signal support for USLE resolution (when applicable). 172
4.20 Relative RMSE versus undersampling ratio δ for all three traffic scenarios in the low SNR regime (SNR = -9 dB for Scenario 1, -3 dB for Scenario 2 and 0 dB for Scenario 3). 173
4.21 Decision tree to help tune the number of measurements M (and thus the sampling rates M_i). 177
5.1 Proposed scenario of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure. 183

List of Tables

1 Distribution of DSP capabilities between local, field-deployed equipment and remote, cloud-like equipment for different system models. 33

2.1 The mean (standard deviation) of the proposed noise variance estimator ([2]'s $AMSL$ in italics). The smoothing window length is $w_l = 30$ samples. 96
2.2 Required SNR (dB) and various occupancy ratios ρ for sample-wise detection with $P_D = 0.99$ and $P_{FA} = 0.01$. 106
2.3 Recommended window lengths for a fraction of NH samples $\nu \leq .1$. 112

4.1 Characteristics of our QPSK-based frames used to simulate the behavior of IoT communications. 141
4.2 Total SNR for the different SNR regimes and traffic scenarios. 143
4.3 Execution times (in seconds) for the different solvers for traffic scenario 1 (low traffic) in the high SNR regime (SNR = 3 dB). 164
4.4 Execution times for the different solvers for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). 166
4.5 Execution times for the different solvers for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). Solvers use the actual signal support for USLE resolution (when applicable). 166
4.6 Execution times for the different solvers for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). 167
4.7 Execution times for the different solvers for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). Solvers use the actual signal support for USLE resolution (when applicable). 168
4.8 Execution times for the different solvers for traffic scenario 1 (low traffic) in the medium SNR regime (SNR = −3 dB). 169
4.9 Best solvers for different traffic scenarios and SNR regimes. 175

List of acronyms

ADC Analog-to-Digital Converter
AMRS Asynchronous Multi-Rate Sampling
AMSL Automated Multiple Shortest Lengths
ANFR Agence Nationale des FRéquences
AP Access Point
AUD Active User Detection
AWGN Additive White Gaussian Noise
BBU Baseband Unit
BER Bit Error Rate
BNIHT Block Normalized Iterative Hard Thresholding
BOMP Block Orthogonal Matching Pursuit
BP Basis Pursuit
BPDN Basis Pursuit Denoising
BTS Base Transceiver Station
BW Bandwidth
CBRS Citizens Broadband Radio Service
CD Cyclostationary Detection
cdf Cumulative Distribution Function
CE Channel Estimation
CF-massive MIMO Cell Free Massive Multiple Input Multiple Output
CIR Channel Impulse Response
CoSaMP Compressive Sampling Matching Pursuit
CovD Covariance-based Detection
CPU Central Processing Unit
CR Cognitive Radio
C-RAN Cloud-Radio Access Network
CS Compressed Sensing
CSI Channel State Information
DB Database
DCT Discrete Cosine Transform
DFT Discrete Fourier Transform
DL Deep Learning
DR Deficient Rank

DSA Dynamic Spectrum Access
DSP Digital Signal Processing
ED Energy Detection
eMTC Enhanced Machine Type Communication
ETU Extended Typical Urban
FC Fusion Center
FCC Federal Communications Commission
FFT Fast Fourier Transform
FISTA Fast Iterative Soft-Thresholding Algorithm
GSM Global System for Mobile Communications
H2H Human to Human
H2M Human to Machine
IF Intermediate Frequency
IHT Iterative Hard Thresholding
IoT Internet of Things
ISFT Iterative Soft Feedback Thresholding
ISM Industrial, Scientific and Medical
KNVE K-means-based Noise Variance Estimator
L1LS ℓ_1-Least Squares
LASSO Least Absolute Shrinkage and Selection Operator
LDACS L-band Digital Aeronautical Communications System
LSA Licensed Shared Access
LTE Long Term Evolution
MCS Multi-Coset Sampling
MFD Matched-Filter Detection
MIMO Multiple Input Multiple Output
MMC Minimal Mean Cluster
mMTC Massive Machine-Type Communications
MOS Metal-oxide Semiconductor
MPPI Moore-Penrose Pseudo Inverse
MRS Multi-Rate Sampling
MSE Mean Square Error
MSL Multiple Shortest Lengths
MWC Modulated Wideband Converter
NB-IoT Narrowband Internet of Things
NB-SS Narrowband Spectrum Sensing
NH Neither Hypothesis
NOMA Non-Orthogonal Multiple Access
NP Neyman-Pearson
NP-hardness Non-deterministic Polynomial-time hardness
NSP Null-Space Property

NUS Non-Uniform Sampling
OFDM Orthogonal Frequency-Division Multiplexing
OMP Orthogonal Matching Pursuit
ONC Optimal Number of Clusters
OSI Open Systems Interconnection
PPPA Predual Proximal Point Algorithm
PSD Power Spectral Density
PU Primary User
QFR Quasi Full Rank
QPSK Quadrature Phase Shift Keying
RD Random Demodulation
RF Radiofrequency
RIP Restricted Isometry Property
RMSE Root Mean Square Error
ROC Receiver Operating Characteristic
ROMP Regularized Orthogonal Matching Pursuit
RRC Root-Raised Cosine
RRU Remote Radio Unit
SC Support Combinator
SD-MRS Spatial Diversity Multi-Rate Sampling
SE Support Estimator
SMRS Synchronous Multi-Rate Sampling
SNR Signal to Noise Ratio
SoI Signal of Interest
SOLAS Safety Of Life At Sea
SP Subspace Pursuit
StOMP Stagewise Orthogonal Matching Pursuit
SU Secondary User
SS Spectrum Sensing
UE User Equipment
UNB Ultra Narrowband
USLE Underdetermined System of Linear Equations
UWB Ultra Wideband
VLSI Very Large Scale Integration
VRS Variable Rate Sampling
WB-SS Wideband Spectrum Sensing

List of notations

$x(\cdot)$	continuous signal
$x[\cdot], x$	discrete signal (vector)
X	discrete Fourier transform of x
$\|\| \cdot \|\|_p$	ℓ_p-norm ($p \neq 0$)
$\|\| \cdot \|\|_0$	ℓ_0-pseudonorm
$\{\cdot\}$	set
$\mathrm{Card}(\cdot)$	number of elements in a set
$\tilde{\Phi}$	dual of frame Φ
$\langle \cdot, \cdot \rangle$	scalar product
$.^T$	transpose
I_n	identity matrix of size $n \times n$
Σ_k	set of all k-sparse vectors
$\ker \cdot$	null space of a matrix
$\mu(\cdot)$	coherence of a matrix
$spark(\cdot)$	spark of a matrix
$\hat{.}$	estimate
$.^\dagger$	Moore-Penrose pseudo-inverse
$.^C$	complement
H_k	hard thresholding operator
$\mathbb{1}_0(z)$	indicator function of the $\{z = 0\}$ set
$\mathcal{H}_0$	null hypothesis
$\mathcal{H}_1$	alternative hypothesis
σ_w^2	power spectrum density of an additive white Gaussian noise
P_D	probability of correct detection
P_{FA}	probability of false alarms
$T_{ED}(\cdot)$	test statistic of energy detection
$T_{MFD}(\cdot)$	test statistic of matched-filter detection
$r_y(n, \nu)$	instantaneous autocorrelation function
$\tilde{r}_y(\alpha, \nu)$	cyclic autocorrelation function
$s_y(\alpha, f)$	spectral correlation function
$\mathcal{N}(\mu, \sigma^2)$	normal distribution
$\mathrm{rank} \cdot$	matrix rank

List of publications of the author during the PhD

International conferences

- E. Selva, Y. Louët and A. Kountouris, "Sparse signal detection with spatial diversity using multi-rate sampling," *2019 URSI Asia-Pacific Radio Science Conference (AP-RASC)*, 2019, pp. 1-4, doi: 10.23919/URSIAP-RASC.2019.8738211.
- E. Selva, A. Kountouris and Y. Louet, "On the Choice of Sampling Rates in Multi-Rate Sampling," *2021 Data Compression Conference (DCC)*, 2021, pp. 370-370, doi: 10.1109/DCC50243.2021.00071.
- E. Selva, A. Kountouris and Y. Louet, "K-Means Based Blind Noise Variance Estimation," *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, 2021, pp. 1-7, doi: 10.1109/VTC2021-Spring51267.2021.9449072.
- E. Selva, S. Ans, Y. Louët and A. Kountouris, "Support Estimation of Frequency-Sparse Signals," *in the submission process*, 2021.

Patents

- Esteban Selva and Apostolos Kountouris. Détermination du bruit dans un signal, 2020.

Résumé en français

Introduction

La forte augmentation présente et à venir du nombre d'objets connectés à l'internet des objets (IdO) n'est pas sans poser de questions quant à la mise à l'échelle de la collecte des messages provenant de ces objets connectés. En outre, plusieurs études mettent en lumière l'encombrement du spectre fréquentiel alloué, et il y a une réelle difficulté à allouer de nouvelles bandes de fréquences à de nouveaux utilisateurs tels que les objets connectés. On sait pourtant que certaines bandes de fréquence sous license sont sous-utilisées par les détenteurs de licence. Cela a donné naissance à l'accès au spectre dynamique (ASD), un cadre établissant une hiérarchie entre utilisateurs primaires détenteurs d'une license sur une bande de fréquences mais qui sous-utilisent la bande qui leur est allouée d'une part, et utilisateurs secondaires, non détenteurs d'une telle license mais souhaitant communiquer tout de même d'autre part. Dans ce cadre, un utilisateur secondaire peut communiquer dans une bande de fréquence allouée mais libre de toute émission, tant que cette communication ne gêne pas l'utilisateur primaire de cette bande. La détection de spectre (DS, en anglais *spectrum sensing*) permet de déterminer si une bande de fréquence donnée est libre de communication ou non. Afin de trouver la meilleure opportunité de communication possible, un utilisateur secondaire peut effectuer une DS-bande large (DS-BL), c'est-à-dire une DS sur une large bande de fréquence. Ceci est au coeur de la radio cognitive (RC), un cadre dans lequel chaque utilisateur acquiert des connaissances sur son environnement radio afin d'optimiser ses capacités de communication.

Dans cette thèse, nous proposons une application proche de l'ASD mais qui en diffère sur plusieurs aspects. Cette application utilise également les préceptes de la RC, y compris la DS-LB, et consiste en la collecte efficace des signaux émis par les objets connectés. De manière générale, la collecte des signaux s'effectue *via* une étape d'échantillonnage. Dans l'industrie, l'échantillonnage est souvent effectué à la fréquence de Nyquist, car cela repose sur des composants électroniques simples et n'implique pas de calculs complexes. Cependant, nous pensons que dans notre situation, l'échantillonnage à la fréquence de Nyquist est inefficace et difficile à mettre en place. Nous travaillons sur des signaux large bande, ce qui implique un taux d'échantillonnage à la fréquence de Nyquist élevé, pour lequel les convertisseurs analogique-numériques (CAN) peuvent être mis en défaut. De plus, nous considérons que les objets connectés émettent sporadiquement, ce qui signifie que le signal reçu est *parcimonieux*, c'est-à-dire principalement constitué de trous (dans les domaines fréquentiel, temporel et/ou spatial). Dans ce contexte, la théorie de l'échantillonnage comprimé (ÉC) semble particulièrement appropriée.

Celle-ci stipule que si un signal est parcimonieux, moins de valeurs du signal peuvent être acquises, et ce sans perte de données. Ainsi, l'ÉC nous permet d'échantillonner des signaux plus efficacement.

Ces éléments nous permettent d'imaginer une infrastructure efficace de collecte des signaux provenant d'objets connectés. Nous nous appuierons sur un schéma d'ÉC intitulé l'échantillonnage multi-taux (ÉMT), ainsi que sur la DS-LB et la RC qui nous permettront d'acquérir des informations précieuses sur l'environnement radio des antennes de collecte.

Nos contributions sont les suivantes. Tout d'abord, nous effectuerons un état de l'art de l'ÉC et de la DS. Ensuite, nous présenterons deux outils que nous avons développé dans le cadre de la RC, à savoir l'estimateur de la variance du bruit sur la base des K-moyennes (EVBK) et l'estimateur de support spectral (ES). Après, nous nous intéresserons au paramétrage de l'ÉMT. Enfin, nous introduirons et évaluerons notre prototype d'infrastructure d'échantillonnage, avant de conclure.

État de l'art

L'ÉC peut être modélisé par un système d'équations linéaires sous-déterminé (SELS)

$$y = Ax, \tag{1}$$

où x est le vecteur d'entrée de taille N (supposé discret et échantillonné à la fréquence de Nyquist), y est le vecteur de mesures (de taille $M < N$) et A est la matrice d'échantillonnage de taille $M \times N$.

Reconstruire le message contenu dans un signal échantillonné grâce à l'ÉC revient à résoudre le SELS (1). Un résultat très connu d'algèbre linéaire stipule qu'un SELS possède une infinité de solutions. Pour trouver la solution $\hat{x}$ qui correspond à l'entrée x, nous utilisons la parcimonie du signal x, qui peut être mesurée par la pseudo-norme ℓ_0. Il existe des garanties théoriques sur la capacité d'un SELS à être résolu, comme la propriété d'isométrie restreinte (PIR, en anglais *Restricted Isometry Property*).

Il existe plusieurs schémas d'ÉC, comme l'échantillonnage multi-coset, le convertisseur à bande large modulée ou l'ÉMT précédemment mentionné. Dans cette thèse, nous nous concentrons sur l'ÉMT. Cet échantillonneur est constitué de L branches parallèles, sur chacune desquelles se situe un CAN échantillonnant de manière uniforme à un taux différent des autres CAN et sous la fréquence de Nyquist dans tous les cas. La matrice A du SELS (1) dépend de L et des taux d'échantillonnage dans chaque branche.

La DS-BL peut nous fournir des informations précieuses quant à la structure du signal d'entrée. En général, les méthodes de DS-BL s'appuient sur la DS en bande étroite (DS-BÉ), appliquée itérativement à un découpage d'une bande large en plusieurs bandes, ou sur l'ÉC, auquel cas le vecteur de mesures y est utilisé pour reconstruire non pas le signal original x, mais pour extraire des informations sur celui-ci (par exemple son support spectral). Cependant, nous estimons qu'il est possible d'utiliser des informations issues de la DS-BL pour résoudre un SELS obtenu par ÉC, plutôt que l'inverse, c'est-à-dire résoudre (même partiellement) un SELS obtenu

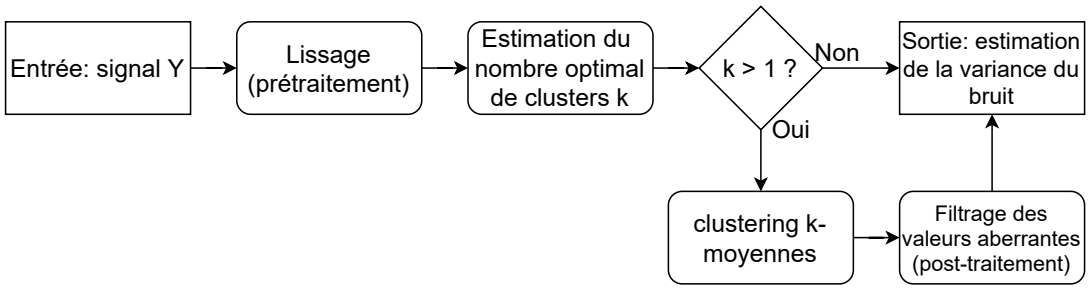


Figure 0.1: Description de l'estimateur de la variance du bruit fondé sur les K-moyennes.

par ÉC pour en déduire des informations relatives à la DS-BL. En particulier, nous allons nous intéresser à l'estimation du support fréquentiel du signal grâce aux échantillons collectés par l'ÉMT, puis nous servir de cette estimation pour simplifier le SELS (1).

Outils pour la radio cognitive

Dans le cadre de cette thèse, nous avons développé deux outils pour l'estimation du support fréquentiel d'un signal parcimonieux, sans information préalable sur le signal reçu (nous supposons que le canal de transmission est un canal bruit blanc gaussien additif, BBGA, de variance inconnue). Le premier outil est l'estimateur de la variance du bruit fondé sur les K-moyennes (EVBK), et le second est l'estimateur de support (ES). Plus précisément, l'estimation du support du signal est effectuée par l'ES, qui prend comme entrée la variance du BBGA calculée par l'EVBK.

Pour les deux outils, nous supposons que le signal reçu est un mélange de signal utile et de BBGA. Le signal utile n'est pas présent dans tout le domaine fréquentiel, alors que le BBGA si. Nous supposons que le signal utile possède une certaine compacité, c'est-à-dire que les valeurs non nulles de signal utile dans le domaine fréquentiel sont concentrées autour d'un nombre limité de points, et qu'il y a des trous relativement larges dans le domaine fréquentiel.

Nous commençons par la présentation de l'EVBK. L'EVBK consiste à séparer les valeurs de signal reçu correspondant à du bruit uniquement de celles correspondant à un mélange signal utile et bruit. C'est une méthode de complexité linéaire en la taille du vecteur d'entrée. La figure 0.1 décrit les différentes étapes de l'EVBK. Ces différentes étapes comprennent un prétraitement qui consiste en un moyennage glissant, une estimation du nombre optimal de clusters grâce au score de silhouette, un clustering grâce à l'algorithme des K-moyennes et un post-traitement qui consiste en le filtrage des valeurs aberrantes. Ensuite, l'EVBK renvoie l'estimation de la variance du bruit.

La figure 0.2 montre les résultats de l'EVBK pour un signal reçu en fonction du rapport signal à bruit (RSB). Le BBGA a une variance connue de 1.0 (unité arbitraire): ainsi, une valeur d'estimation de la variance du bruit proche ou égale à 1.0 correspond à une bonne performance de l'EVBK, qui peut alors être considéré comme non-biaisé. Même pour des signaux peu parcimonieux, on voit qu'on peut estimer la variance du bruit de manière très satisfaisante pour un régime de RSB élevé (RSB > -7 dB pour un taux d'occupation spectrale de 20%, RSB > 4

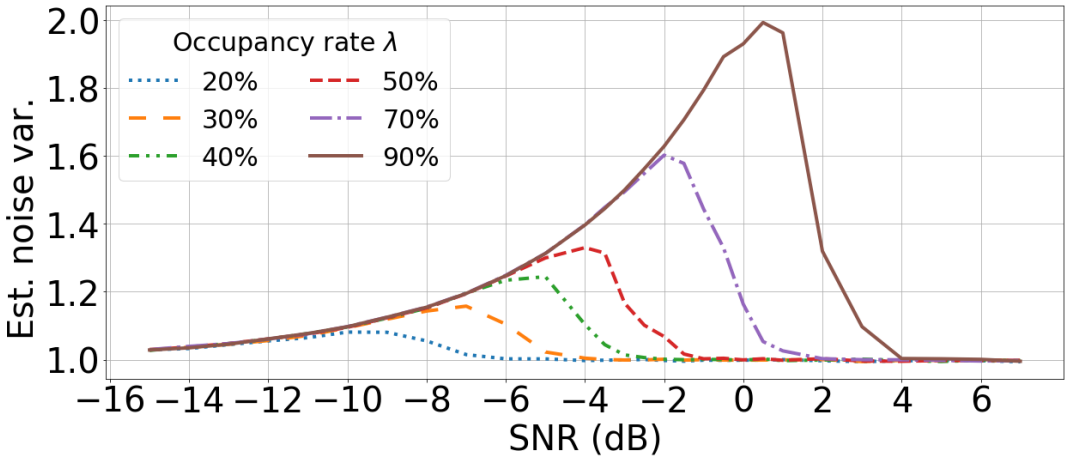


Figure 0.2: Estimation de la variance du BBGA en fonction du RSB pour différents taux d'occupation du spectre fréquentiel.

dB pour un taux d'occupation spectrale de 90%).

L'EVBK a fait l'objet d'une publication en conférence internationale avec relecture des pairs [3] ainsi que d'une demande de brevet auprès de l'Institut National de la Propriété Intellectuelle [4].

Nous passons ensuite à la présentation de l'ES. L'ES repose sur la détection d'énergie (une technique de DS), appliquée à chaque valeur de la transformée de Fourier Y du vecteur de mesure y. Chaque valeur $Y[i]$ contient soit du bruit uniquement (hypothèse nulle $\mathcal{H}_0$), soit un mélange signal-bruit (hypothèse alternative $\mathcal{H}_1$). Ainsi, la détection d'énergie dans chaque valeur $Y[i]$ consiste en une décision binaire: soit $\mathcal{H}_0$ est décidé, soit $\mathcal{H}_1$. La décision se fait en comparant l'énergie contenue dans $Y[i]$ à un seuil dépendant de la variance du BBGA, d'où l'intérêt de l'EVBK précédemment introduit.

On dit qu'une détection correcte (resp. une erreur de type I) se produit quand $\mathcal{H}_1|\mathcal{H}_1$ (resp. $\mathcal{H}_1|\mathcal{H}_0$) est décidé. La théorie de Neyman-Pearson propose un cadre dans lequel la probabilité de détection P_D est maximisée pour une probabilité d'erreur de type I P_{FA} donnée. Seulement, les expressions explicites (comme celle liant P_D à P_{FA}) proposées dans le cadre de la théorie de Neyman-Pearson reposent sur le théorème central limite, inapplicable dans notre cas. Pour l'ES, nous revisitons donc ces équations et proposons une réécriture de celles-ci sans l'utilisation du théorème central limite. La figure 0.3 illustre la courbe ROC (*Receiver Operating Curve*) de l'ES pour différentes valeurs d'énergie du signal utile (ces valeurs sont reliées au RSB et au niveau d'occupation spectrale). On voit que plus l'énergie du signal utile est grande, plus la courbe ROC est proche du coin supérieur gauche du plan $P_D - P_{FA}$, indiquant une meilleure performance de l'ES.

Nous mettons également au point une amélioration de l'ES, qui se matérialise par un prétraitement consistant en une étape de fenêtrage similaire à celle de l'EVBK. Celui-ci permet d'améliorer la performance de l'ES, à énergie du signal utile constante.

Grâce à l'EVBK et à l'ES, nous pouvons estimer le support fréquentiel du signal utile dans un vecteur de mesure. Désormais, nous abordons le paramétrage de l'ÉMT, le schéma d'ÉC

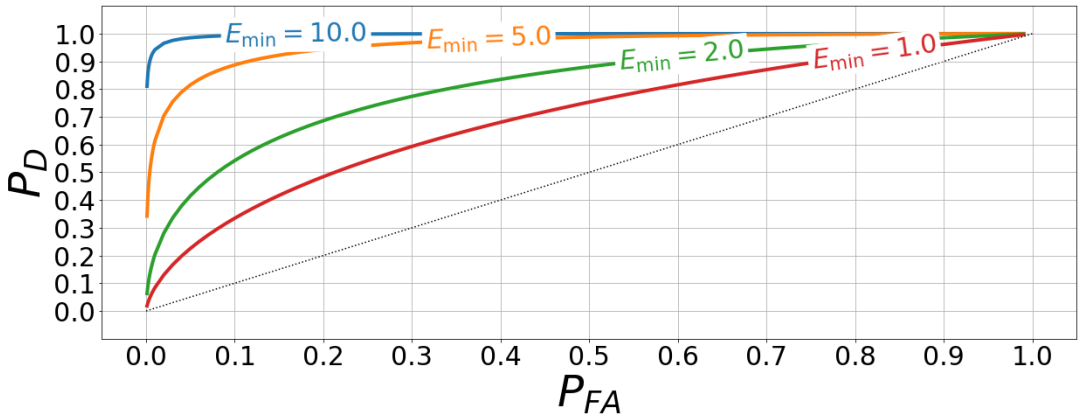


Figure 0.3: Courbe ROC de l'ES sans fenêtrage pour différentes valeurs d'énergie $E_{\min}$ (unités arbitraires).

utilisé au cours de cette thèse.

Paramétrer l'ÉMT

Afin de pouvoir reconstruire le SELS (1), il est important de bien régler les paramètres de l'ÉMT. Ceux-ci sont le nombre de branches L et les taux d'échantillonnage ν_i $(1 \leq i \leq L)$. Dans chaque branche, nous définissons l'entier $M_i = \Delta\nu_i$ (resp. $N = \Delta\nu_{Nyq}$) comme le nombre d'échantillons acquis dans la branche i (resp. dans une branche hypothétique échantillonnant à la fréquence de Nyquist ν_{Nyq}) pendant la durée d'acquisition Δ. Ainsi, nous étudions le rôle du nombre de branches L et des M_i.

Dans cette section, notre contribution est double. D'abord, à travers le théorème 0.1, nous établissons un lien entre L, les M_i et le rang de la matrice d'échantillonnage A, noté rg A. Ensuite, pour quantifier la capacité d'un ÉMT à échantillonner de manière à ce que le SELS (1) soit résoluble, nous proposons de remplacer la métrique communément admise du nombre de mesures $M = \sum_i M_i$ par rg A.

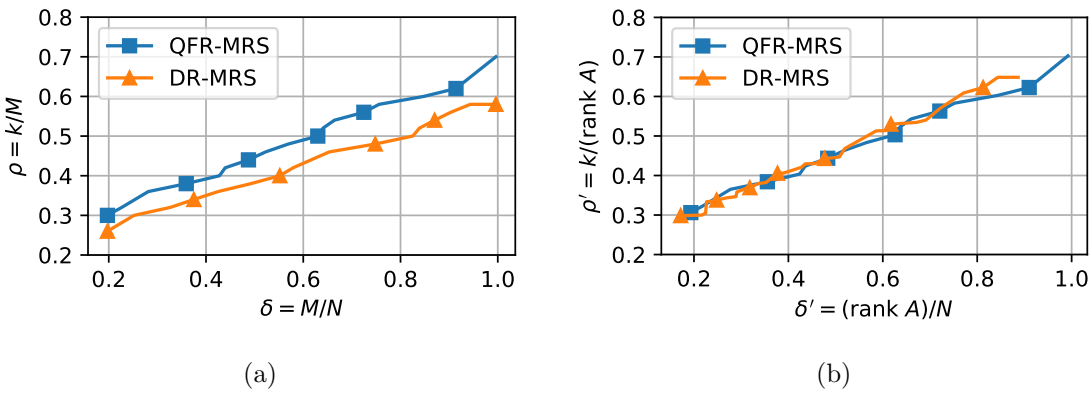
Théorème 0.1 (Limite supérieure du rang de la matrice)**.** *Soit A la matrice d'échantillonnage de taille $M \times N$ d'un ÉMT à L branches. Alors*

$$rg\ A \leq M - (L - 1). \tag{2}$$

Le cas d'égalité est atteint si et seulement si les M_i sont deux-à-deux premiers entre eux.

À la lumière de simulations, il apparaît que rg A est plus à même de caractériser la capacité d'un ÉMT à recouvrer un signal échantillonné sous la fréquence de Nyquist que le nombre de mesures M. La figure 0.4a montre la transition entre les phases[1] de succès et d'échec de résolution de SELS dans le plan $\rho - \delta$, où $\rho = k/M$ est le taux d'occupation spectrale et où $\delta = M/N$ est le nombre de mesures normalisé. Au dessus de chaque courbe, un ÉMT ne peut

[1]Ici, le terme "phase" se comprend comme une région du plan $\rho - \delta$ où la performance de reconstruction est uniforme.



(a) (b)

Figure 0.4: Transitions entre phases de succès et d'échec pour un (a) nombre de mesures normalisé; (b) rg A normalisé. QFR-MRS = ÉMT avec M_i premiers entre eux deux-à-deux. DR-MRS = ÉMT avec M_i non deux-à-deux premiers entre eux.

pas espérer résoudre un signal non bruité ayant k éléments non nuls avec l'algorithme de minimisation de norme ℓ_1 CVXOPT. En dessous de chaque courbe, un ÉMT acquiert suffisamment d'informations pour une parfaite reconstruction du signal. La courbe étiquetée "QFR-MRS" (resp. "DR-MRS") correspond à des M_i deux-à-deux premiers entre eux (resp. non deux-à-deux premiers entre eux). On voit que pour toute valeur de δ, la performance de l'ÉMT avec taux premiers entre eux est supérieure à celle de l'ÉMT avec taux non premiers entre eux. En outre, la performance ne dépend pas que de M, mais aussi du fait que les taux soient ou non premiers entre eux. En revanche, en remplaçant M par rg A, les courbes se superposent presque parfaitement (figure 0.4b), indiquant que le rg A est plus à même de caractériser la performance de reconstruction d'un ÉMT.

Ces travaux ont fait l'objet d'une publication en conférence internationale avec relecture des pairs [5]. À présent, nous nous intéressons à la combinaison de tous les blocs étudiés jusqu'à présent au sein d'une infrastructure d'échantillonnage fondée sur l'ÉMT et tournée vers l'IdO.

Une infrastructure d'échantillonnage tournée vers l'IdO fondée sur l'ÉMT

Nous proposons de donner à notre proposition d'infrastructure d'échantillonnage l'objectif d'échantillonner efficacement des signaux provenant d'objets connectés. Pour ce faire, l'infrastructure se doit d'échantillonner ni trop lentement, pour ne pas perdre d'information provenant des objets connectés ni trop vite, ce qui causerait un gâchis d'énergie et de mémoire. La nature de l'environnement radio étant susceptible d'évoluer au cours du temps, l'infrastructure doit également être pourvue de facultés d'adaptation à cet environnement radio.

Notre infrastructure se compose de plusieurs blocs fonctionnels, tels qu'une interface analogique, un ÉMT à L branches, L EVBK parallèles, L ES parallèles, un bloc de combinaison de supports estimés partiels, une base de données de stockage d'échantillons et un bloc

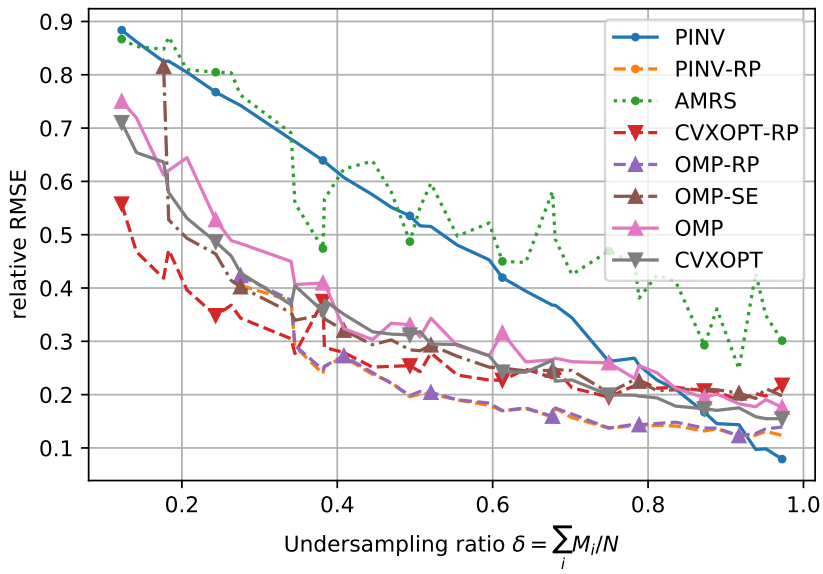


Figure 0.5: Racine de l'erreur quadratique moyenne relative en fonction du nombre de mesures normalisé pour un faible volume de trafic et un RSB élevé (RSB = 3 dB).

de résolution de SELS (aussi appelé solveur).

Pour tester les différentes étapes de notre infrastructure, nous mettons au point un générateur de trafic IdO, où les trames des objets connectés sont représentés par des signaux de forme QPSK filtrée. Ces trames sont soumises à un canal de Rayleigh avec addition de BBGA. Différents volumes de trafic, représentatif de différents scénarios de fonctionnement, sont implantés.

Nous testons ensuite chaque étape de notre infrastructure, et ce pour différents nombres de branches L et valeurs de M_i et différents RSB.

Concernant le bloc de reconstruction, en fin de la chaîne de traitement, nous implantons des solveurs représentatifs de différentes familles d'algorithmes de résolution de SELS.

De manière générale, plus le niveau d'occupation spectrale est bas, plus le nombre de mesures M (quasiment égal à rg A dans le cas où les M_i sont deux-à-deux premiers entre eux) peut être réduit tout en conservant une capacité de reconstruction du signal satisfaisante. Ainsi, la figure 0.5 représente l'erreur entre le signal reçu, échantillonné puis reconstruit et le signal reçu échantillonné à la fréquence de Nyquist, en fonction du nombre de mesure normalisé pour un faible volume de trafic pour un RSB élevé. Le nombre de branches de l'ÉMT est $L = 4$ branches. Les différentes courbes représentent les différents solveurs testés. Il y a un compromis à faire entre haute qualité de reconstruction et faible nombre de mesures; néanmoins il est possible d'avoir un nombre de mesures bas et une erreur relative basse. Par exemple, le solveur OMP-RP permet d'avoir une erreur relative assez faible, de 0.2, avec un nombre de mesures $N = M/2$.

Les conditions d'environnement radio ayant un impact sur la performance de l'infrastructure d'échantillonnage sont principalement le RSB et le niveau d'occupation spectrale. En fonction de ces paramètres, certains solveurs sont à privilégier. De plus, il convient d'adapter les paramètres de l'ÉMT (principalement son nombre de mesures, donc ses taux d'échantillonnage)

en fonction des conditions d'environnement radio. Nous proposons donc des pistes afin de rendre l'infrastructure d'échantillonnage adaptative.

Conclusion et perspectives

Nous revenons sur les travaux effectués au cours de la thèse et les enseignements qu'il est possible d'en tirer. Simultanément, nous offrons des perspectives d'amélioration de l'infrastructure d'échantillonnage proposée. Ensuite, nous explorons la perspective d'une extension de plus grande ampleur de l'infrastructure d'échantillonnage. Il s'agit de l'extension d'une architecture mono-site, où il y a une unique antenne située sur un seul site, vers une architecture multi-sites, où des antennes situées à des endroits géographiquement distincts échantillonnent les signaux reçus avant regroupement des échantillons en un centre de fusion pour traitements ultérieurs. Cela permettrait d'exploiter la diversité et la parcimonie spatiales, mais plusieurs défis se présenteraient alors.

General Introduction

0.1 History and context of wireless transmission-collecting infrastructure

The 1912 sinkage of – then commonly considered unsinkable – *RMS Titanic* astonished the people of the early 20th century. As a response, the Safety of Life at Sea Convention (SOLAS)[6] was ratified in 1915. This treaty contained provisions for the safety of ships, among which a chapter on radio communications. In particular, Article 33 specifies that the larger ships be "required to be fitted with a radiotelegraph installation" with a "continuous service" in order to receive potential distress signals from other ships. Even though the service was not meant to process the received signals any further – except by taking action and rescue the ship in distress –, nor to save them for future use, this constitutes the first example of an ad-hoc, point-to-point wireless network with a systematic harvesting of radio transmissions, a mere 20 years after the first successful wireless transmission experiments by Guglielmo Marconi. By essence, this primitive network was completely decentralized.

Later on, two paradigms for radio communications emerged. On the one hand, point-to-point radio communications remained used mainly for marine, aviation and military purposes. On the other hand, radiodiffusion, based on downlink broadcasting, became popular amongst the general public. The sophistication of the transmitters and receivers, as well as the growing number in users, led the US Congress to pass the Radio Act of 1927 [7], which required broadcasters to obtain a license to operate on a frequency band. Aside from radio broadcast recorders, the radio waves transmitted by licensees were not stored nor processed beyond the pleasure of listening to a radiophonic program.

For the general public, duplex wireless communications remained marginal until the 1990s.

0.1.1 Centralized versus decentralized signal processing

The cellular model, backbone of the 1990s' boom of wireless communications Approximately eighty years after *Titanic* sank, the major breakthrough that drove the massification of wireless communications is the introduction of digital signal processing (DSP) [8]. It was carried by then-recent advances in hardware technology: Very Large Scale Integration (VLSI) of metal-oxide semiconductor (MOS) transistors [9]. Partly because of cheaper end-user equipment, there was an increasing demand on wireless network infrastructure. It was accomodated with more offer of spectrum resource, thanks to more spectral-efficient digital modulations.

Around the turn of the 1990s, the cellular model, initially conceived by Bell Labs in 1947 [10], was also widely implemented. The Global System for Mobile Communications (GSM) was the first digital cellular network standard. In GSM, a geographical unit called a cell is served by a Base Transceiver Station (BTS); phones (known as User Equipments, UE) present in the cell communicate in duplex mode with the cell's BTS. The geographical division of land was driven by the possibility of reusing frequency channels in different cells. For uplink communications, DSP takes place at the BTS, and the processed information is then transmitted to the core network.

Unity makes strength (and scaling reduces costs) Closer to our day and age, there has been a trend towards (more) centralized DSP. This shift from decentralized to centralized DSP is for instance exhibited in the Cloud-Radio Access Network (C-RAN) architecture [11]. In C-RAN, several Remote Radio Units (RRU) are connected through a Fronthaul link to a pool of Baseband Units (BBU). In previous generation networks, the BBU and RRU would have been combined to form a BTS. This is done in an effort to mutualize the DSP in the BBU pool while lowering costs of RRU deployment (thus allowing to deploy more RRUs, and more service for the end users, at constant cost). This centralization of DSP is analog to what has been witnessed in cloud computing in the recent years, with the massification of remote computations with the aim of reducing costs through scaling effects.

The Next Episode: Collaboration is Key The next step in the race towards higher data-rate, lower latency communications is the development of Cell-Free (CF) massive Multi Input Multi Output (MIMO). Massive MIMO is a technology developed to provide more connectivity to end users. It relies on precise user-targeted spatial multiplexing, which increases the system throughput. However, a bottleneck in massive MIMO is inter-cell interference [12]. To suppress this interference, the cellular network model is challenged by cell-free network models. In CF massive MIMO, UE data is transmitted to and from several Access Points (AP) connected to a Central Processing Unit (CPU), which performs DSP on signals jointly received by the different APs. This collaboration between the different APs makes it possible for CF massive MIMO to take advantage of added macro-diversity, and the characteristics of the communications system (such as the channel gain) are improved. Since the DSP is performed at the CPU (except for channel estimation), APs do not need strong computing capabilities. This centralization of DSP is somewhat analog to cloud computing, where a user with high computation requirements can use remote ("cloud") scalable processing resources instead of a local, high-end processing system.

Keep calm and become aware of your radio environment Contrary to the centralized DSP trend, the paradigm of the Cognitive Radio (CR), introduced by Joseph Mitola III in 1999 [13], advocates for a different type of radio. Through Spectrum Sensing, the process of monitoring the surrounding radio environment for signal identification, a CR is aware of its spectrum environment and can adapt the characteristics of its communications through software

System Model	DSP performed... locally (at BTS/UE/RRU/AP/equiv.)	remotely (at BBU/CPU/equiv.)
Cellular	●●●●●	○○○○○
Cognitive Radio	●●●●○	●○○○○ (distributed)
C-RAN	○○○○○	●●●●●
Cell-Free	●○○○○	●●●●○

Table 1: Distribution of DSP capabilities between local, field-deployed equipment and remote, cloud-like equipment for different system models.

reconfiguration. In this scenario, DSP is mainly carried out at the UE itself. This paradigm has come out as a solution to the inefficiency of frequency spectrum use. Indeed, while the spectrum is utilized to its maximum capacity in some frequency bands (such as the cellular bands), other bands are underutilized (for instance, amateur radio or paging bands). Hence, a software-reconfigurable radio would be able to operate on bands not used at a given time/space. Since the CR paradigm is in essence decentralized, it is well-suited for unplanned, ad-hoc networks. Crowd-sourcing initiatives, like Electrosense [14], have taken advantage of the CR paradigm for spectrum data collection and analysis. However, since its inception, the CR paradigm has not reached widespread commercial deployment for end-user communications, and the IEEE 802.22 standard aiming at using TV white space for CR communications has entered a state of hibernation in July 2019.

Table 1 summarizes the distribution of the DSP for the aforementioned infrastructure architectures.

0.1.2 The Internet of Things: reshuffling the cards of service requirements

New requirements for new use cases The Internet of Things (IoT) is a network where diverse devices (things) are interconnected and able to send and receive data without human interaction. IoT communication requirements are typically quite different from those of humans: where (some) people strive for instant 4K video streaming anytime, anywhere, a IoT device such as a connected sensor might communicate as little as one 12-byte payload a day. On the other hand, IoT use cases raise other constraints that make current and future wireless communications standards (LTE, Bluetooth, CF massive MIMO...) irrelevant and inefficient. For instance, the complexity of the LTE protocol (overhead, back-and-forth communication for resource allocation, and so on) is incompatible with the necessity for the device to save its energy (to keep maintenance costs low, the battery needs to be changed as rarely as possible).

Ideal and reality The ideal scenario for an IoT device would be to transmit its message with little or no preprocessing and overhead, without prior channel listening or resource allocation. A radio unit would switch itself on and collect data transmitted according to a schedule predefined by the device(s) and the radio. There would be one communication standard and no

interference.

Of course, predefined schedules are hardly scalable, and the perspective of a single standard seems unlikely at the present time. There is also the problem of interference, between devices themselves first, and between devices and other non-IoT wireless uses. This problem is closely related to that of the frequency spectrum scarcity: most if not all frequency spectrum suitable for wireless communications is already allocated, leaving little room for new agents and use cases.

On the data-collecting infrastructure Since scheduling seems to be too strong a hypothesis, it is fair to consider that a device can transmit its message at virtually *any* time – based on some inner heuristic or algorithmic output – and that someone who wants to get the message needs to scan the radio environment at *all* times. Furthermore, the (current) multiplicity of IoT communications standards and the will to keep IoT costs low – necessary for widespread deployment of the technology – leads to considering a unique, multi-standard, inexpensive-to-deploy radio data-collecting infrastructure. An argument for having a unique multi-standard data-harvesting infrastructure rather than one infrastructure per standard is that a critical mass of data is needed to make a collecting infrastructure profitable: the more low duty-cycle IoT devices one can collect information from (at constant cost), the better. A unique infrastructure also obviously lowers the barriers to entry for new agents with new modulation standards fitted to their own specific communication needs.

Best of both worlds: centralized DSP combined with local intelligence IoT's specific communication requirements call for new solutions for data-collecting infrastructure. Several principles of the CR paradigm seem particularly fitted to new IoT use cases. For instance, based on the knowledge of spectral occupation by the IoT devices, a radio environment-aware (or "intelligent") radio unit could routinely and adaptively configure its sampling characteristics, such as its sampling rate(s) or bandwidth. It does not matter whether this environment analysis takes place locally at the radio unit or remotely at some processing unit, as long as it is actually performed and as long as the radio unit can self-reconfigure accordingly.

Other innovations present in next-generation technologies like CF massive MIMO can be put to good use in IoT data-collecting infrastructure. The most straightforward such innovation is collaboration between spatially distinct radio units. Since collaboration aims at improving the efficiency of communications, it can be beneficial to use cases as diverse as instant, high data rate communications (the use case targeted by CF massive MIMO) or thousands of low duty-cycle IoT devices deployed in vast areas. We will define the level of collaboration and its implications on the system design later on.

0.1.3 Conclusion

On the one hand, user-oriented wireless communications are being driven towards higher data rates, lower costs and lower latency; this is (in part) made possible by a shift from local, decentralized DSP to remote, centralized DSP. On the other hand, the particularities of IoT (many

cheap, low-duty-cycle, energy-constrained devices deployed in extremely diverse environments) reshuffle the cards and call for new data-collecting principles. How can a technology-agnostic infrastructure efficiently collect low-overhead data sent irregularly in time, space and frequency by thousands of devices? This PhD thesis aims at providing answers to this vast question.

0.2 Stakes of the present study

0.2.1 Preliminary considerations: scope of the study

At the PHY layer of the OSI model, acquisition of a digitally-modulated signal consists in three consecutive steps:

1. Analog preprocessing (amplification, filtering);
2. Sampling *i.e.* the transform of a continuous signal into discrete samples;
3. Demodulation *i.e.* the operation of converting signal realizations (samples) into symbols and bits.

The symbols then follow a processing chain whose aim is to extract an actual exploitable payload containing the user or application data.

Out of these three steps, we will focus on what we believe is the cornerstone of digital communications, the most universal and most crucial step: sampling. Therefore, our main objective will be the obtention of exploitable samples by upper stack layers. Since we wish to develop a technology-agnostic platform, we will not favor a particular modulation scheme over another. However, for didactic purposes, we may use one or several common IoT modulation schemes to illustrate our findings.

The huge quantity of field-deployed IoT devices entails massive DSP on the uplink, on which we focus throughout this document. The downlink also has its shares of interesting problems but of a quite different nature, which are beyond the scope of the present thesis.

0.2.2 Efficient sampling and how we can do it

We now introduce key concepts lying at the heart of this thesis. First, we delve into the structure of the considered signals and explore the available levers to efficiently sample them. This leads us to take an interest in Compressed Sensing (CS), a sampling paradigm which allows, in certain conditions, to losslessly lower the sampling rate. Afterwards, we turn our attention to Spectrum Sensing (SS), a discipline central to the CR paradigm, and identify its synergies with CS. Finally, we explore spatial diversity, understood as a way to enhance the capabilities of a sampling infrastructure.

The considered signals of interest As mentioned earlier, IoT devices may transmit over wide space/time/frequency ranges: we will have to monitor a wide portion of the frequency spectrum. Since the sizes (in bits) of the transmitted messages are low (up to a few kilobits in

many applications), the signal transmitted by a single device typically does not extend over a wide time-frequency span. Furthermore, low duty cycles mean that a single device will have a very small occupation of the time-frequency spectrum. Even though devices can be numerous in a given area, this combination of factors leads us to make a reasonable assumption on the received signals: we will concentrate our study on (very) *sparse, wide-bandwidth, multi-band* signals. In the basic meaning of the word, sparsity means that at any given time, a very small portion of the frequency spectrum is occupied by device transmissions. This is materialized by a power spectrum density (PSD) consisting mostly of zeros. This mostly-null PSD corresponds to an ideal, noiseless case. In practice, communications undergo propagation in channels with various impairments, such as thermal noise, path-loss, shadowing or multi-path fading. As a result, the PSD of the received signal has no actual zero values and is, formally speaking, no longer sparse unless we consider some form of "information-carrying" energy on a band. A strategy to adapt sparse PSD signals mechanisms to noisy scenarii is the following: instead of dismissing only the zero values of the PSD, one retains the values above some noise-related threshold [15].

The fundamental theorem of sampling and why we need to go beyond The historic Nyquist sampling theorem (first formulated by Kotelnikov in 1933 [16]) states that a bandlimited signal with no frequency component above f_{max} can be perfectly reconstructed from samples taken uniformly at a rate of $2f_{max}$ (a rate further referred to as the Nyquist rate). Uniform sampling is rather easy to perform; moreover, the reconstruction of the original signal from such samples is facilitated by convenient interpolation formulas. This theorem has been the cornerstone of analog-to-digital conversion ever since its introduction and remains the industry-standard way to sample signals, up to the present day.

However, it is not well-suited for sampling wideband sparse signals. Indeed, every Analog-to-Digital Converter (ADC), the electronic component used for sampling a received signal, has an upper limit on sampling rate [17], typically lower than the Nyquist rate for a wideband signal. Even if ADC technology improves, sampling sparse signals at the Nyquist rate is inefficient, for doing so takes far more samples to reconstruct each narrow IoT transmission than necessary. Inefficiency in this domain results in two problems: an energy problem, since high rate ADCs consume a lot of energy, and a volume-of-data problem, since the enormous quantity of collected samples would quickly take a lot of memory. Furthermore, processing more samples demands higher computational resources. While storage is not a problem in the case of real-time sample stream processing, we might want to store samples temporarily for on-request further processing.

As a consequence, we need to consider sub-Nyquist Sampling, the art of sampling (in a reconstructible manner) below the Nyquist rate.

Compressed Sensing and a theoretical lower-bound Sub-Nyquist Sampling, Compressed Sampling or Compressed Sensing (CS), was introduced by Candès, Tao, Donoho et al. in 2006 [18, 19]. The core principle behind CS is that a signal with the property of sparsity can be completely described by a number of coefficients (sometimes much) smaller than the number of

time-domain samples that would have been taken by sampling the signal at the Nyquist rate. These coefficients are not necessarily the acquired time-domain samples but are to be understood as values representing the signal in some basis. Collecting these information-containing coefficients can be done using various sampling methods that have been proposed in the past years. These methods will be thoroughly presented in Chapter 1; yet in this thesis we will focus on one method in particular, called Multi-Rate Sampling (MRS) [20]. MRS relies on the combination of observations of the signal by different sampling branches, each branch sampling uniformly at a rate well below the Nyquist rate[2]. In other words, MRS possesses the advantage of relying on uniform sampling – at a rate lower than Nyquist – as an underlying mechanism. This emphasis on uniform sampling is consistent with our will to deploy a cheap infrastructure, since uniform sampling prevails in the industry.

A key result in CS is the Landau lower-bound on sampling rates [21]. It links the minimum sampling rate for which information can be reconstructed to the Lebesgue measure of the support (in some basis) of the signal. As is often the case, this result is theoretical and can be approached only in a limited number of scenarii.

In this study, the overall sampling rate will typically be bounded by the Landau rate (lower-bound) and the Nyquist rate (upper-bound). Since our aim is to reduce the sampling rate and number of collected samples – under the constraint that we must be able to reconstruct the signal –, we will strive to approach the Landau rate. Since the Landau rate depends on the level of spectrum occupancy of the signal, this level must either be modeled or measured/estimated at some point, using Spectrum Sensing techniques.

Spectrum Sensing and signal support detection Spectrum Sensing (SS) was introduced in the CR paradigm as a enabler for Dynamic Spectrum Access (DSA). DSA relies on a hierarchical model [22] where wireless network users are divided into two categories: the Primary Users (PU), who enjoy unlimited, anytime access to a given set of licensed frequency bands (typically allocated either free of charge or for a fee by the regulator), and the Secondary Users (SU) who are not granted this access but can operate on the same bands as the PU if these bands are not used and if no interference is provoked by SU activity. In this context, SS helps determine whether PU signal is present or absent on a given band. SS methods vary in complexity, efficiency and necessary prior knowledge about the signal to detect, and mostly belong to three detection families: energy detection, cyclostationary feature detection and matched filtering [23].

Because SUs constantly need to look for new available bands, wideband SS has attracted attention in the past years. CS is, for the aforementioned reasons, an enabler of wideband SS [24]. Understandably, the main objective of wideband SS has been spectrum hole identification for SU communications: there was no need to reconstruct the signal transmitted by PUs. Yet, if we move away from the DSA use-case and focus on collecting the samples of this existing

[2]We restrain our attention to the situation where the total sampling rate, or overall sampling rate, defined as the number of measurements taken across all branches during one second, is also below the Nyquist rate. However, this is not mandatory, and the total sampling rate of the MRS can be equal to or above that of Nyquist.

radio traffic, it becomes clear that wideband SS can be used for signal support detection, or support recovery. For some CS schemes, like MRS, support recovery is particularly useful as it provides a strong clue for signal reconstruction (materialized in the reduction procedure of MRS). In particular, support recovery allows to compute the level of spectrum occupancy, on which the Landau rate depends, as mentioned previously. However, to the best of our knowledge, precise support recovery in noisy scenarii for the purpose of signal reconstruction has not been undertaken so far.

Exploit spatial diversity and minimize redundancy Any signal that propagates in the medium is subject to various channel impairments. If transmissions from a device face difficult propagation conditions, the Signal-to-Noise Ratio (SNR) drops and the transmitted messages are lost. Collaborative fusion of samples taken by spatially distributed radio units has been proposed as a solution for this problem [25]. Indeed, the transmissions from a device to the different radio units undergo propagation in independently-fading channels, increasing the odds that the original transmission (or its spectral support) can be recovered. This is an example of the gains made possible by spatial diversity. Of course, because the samples taken at different sampling sites need to be forwarded for reconstruction, spatial diversity exploitation incurs overhead: there is a trade-off to be found between cooperation overhead and signal recovery gains [26].

Under good propagation conditions however, neighbor reception sites can suffer from redundancy. For instance, imagine that a message is transmitted by a device and that the observations (samples) made by a single reception site suffice to extract the useful information present in the message. This renders the neighboring reception sites useless (for this particular device and message). Put differently, this makes sampling by neighbor reception sites redundant. An easy way to mitigate this inefficiency is to switch off redundant sampling sites. On the opposite, a finer approach is to exploit this redundancy, which is more desirable when the totality of sites and redundancies are considered. For example, this can be done by lowering the sampling rate at each site and make the neighbor sites collaborate to obtain an exploitable signal, an idea developed in [27]. Either way, when designing a sampling system, it is valuable to make the sampling sites (re)configurable, so that they can either exploit spatial diversity for improved signal recovery (from transmission impairments) or scale their sampling parameters so as to reduce unwanted redundancy [28].

Centralized or decentralized DSP? In a multi-site sampling infrastructure, samples gathered by the reception sites need to be processed both in a local, individual fashion – to become aware of the local radio environment – and in a collective fashion – to combine the samples and recover the original signal. Collective processing can be done in a distributed manner or a centralized manner, but in our scenario, centralized processing in a Fusion Center (FC) seems more adequate, as we will see in the following section. In short, DSP is performed both locally and remotely.

0.2.3 Prototype of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure: use case of this PhD thesis

The introduction of the stakes and underlying key concepts of this thesis allows us to set the ground for a prototype of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure. This is the prototype of a solution that a telecom operator like Orange (the sponsor of this PhD program) could offer to a business client.

Figure 0.6 provides a vision of the system that includes all the stakeholders to the entire sampling system. This infrastructure consists in local sampling sites (later referred to as cognitive radios, CR) and a fusion center (FC). In our scenario, the client rolls out their IoT devices (later referred to as end devices) over a specific geographical area. The devices live their lives and transmit data at their discretion. Our proposed infrastructure samples the radio environment and performs detection using Spectrum Sensing techniques: if a signal is detected, the samples are kept and stored, otherwise they are dropped. In a simplified vision, to obtain the end devices' data, the client queries our infrastructure by sending a request containing a date and time interval, a frequency range and an optional geographical area. The infrastructure then responds with the samples matching the parameters of the query. If no sample matches the query, an empty response is sent to the client.

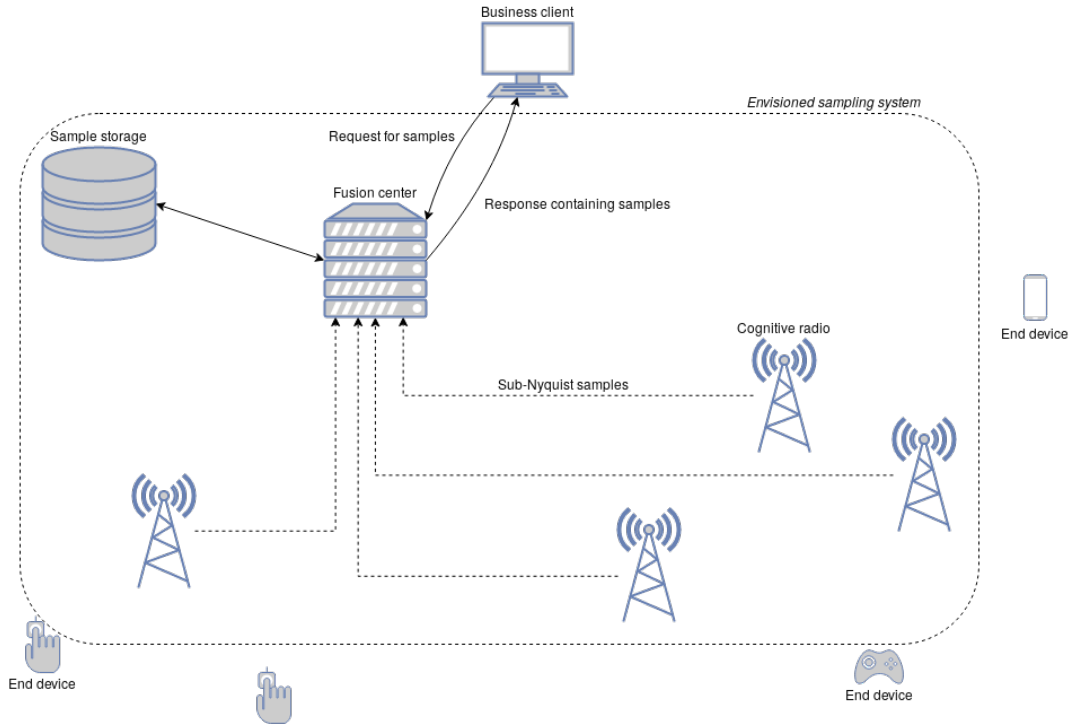


Figure 0.6: Proposed scenario of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure.

Choice of the sub-Nyquist sampling scheme We have chosen to use our proposed extension to the MRS, called the Spatial Diversity Multi-Rate Sampling (SD-MRS) scheme. This extension of MRS seems particularly suitable for fast adoption of our system. Indeed, the hard-

ware in the CRs (the point of entry of end devices' data in the sampling system) is fairly simple and virtually similar to what is embedded on current smartphones. There is no requirement for new ultra-specific hardware components, thus lowering the deployment cost of an SD-MRS based solution. Moreover, SD-MRS is adaptive, resilient and effective.

The role of the CR A CR implements a reception architecture (reception antenna, analog preprocessor, sampler). It can embed one or several sampling branches. Each sampling branch is clocked to a given sampling rate. The samples of each branch undergo a (digital) preprocessing stage that consists in two steps. First, a decision on the presence (or absence) of any actual information in the samples is taken. Second, if presence of information is decided, the location in the wideband spectrum of the sub-bands that contain the information (i. e., the signal support) is estimated. Samples where information is believed to be present are timestamped and sent to the FC via a backhaul link; the other samples are dropped. To keep CR computations light, the tests are based on energy detection. This is facilitated by the fact that sampling below the Nyquist rate does not decrease the SNR, usually a crucial factor that decides the fate of energy-based detection. However, since the samples are taken more distantly in time, the acquisition time for a given number of samples is higher. This reduced time resolution can be detrimental to short messages. A failsafe is to make the statistical tests more tolerant: since a type I error corresponds to a sample containing information dismissed as a sample without information, we just want as little type I errors as possible, and can act accordingly on the parameters of the test.

The role of the FC The first function of the FC is to interact with the Sample Storage unit. Upon reception of the samples sent by the CRs, the FC labels the samples and store them in a sample database. When the client queries the sampling system, the FC retrieves the samples corresponding to the query and combines them. The combination stage occurs in two steps: first, the actual signal support is determined (based on the signal support estimations done locally by the CRs), second, the signal of interest itself is recovered. Note that tight synchronicity between the different CRs is a plus, albeit not strictly necessary. Finally, the FC sends the recombined samples to the client.

Performance measurement The performance objective that determines some quality of service of the system will have to be defined between the infrastructure manager (typically the telecom operator) and the client, based on some appropriate figure of merit. A candidate for the figure of merit is the average sample error rate. For instance, the client may have a requirement such as "I want to gather the correct samples[3] more than 99.9% of the time, and a failure from the telecom operator to provide the remaining 0.1% correct samples is tolerable". The bit error rate (BER) could be another figure of merit, but it depends on the modulation scheme used for the communication which we do not *a priori* know.

[3] A correct sample, in a noisy setting, could be a sample that is not or marginally altered by sub-Nyquist sampling, regardless of phenomena outside of the scope of the sampling infrastructure *e.g.* noise-induced impairments.

Here is a non-exhaustive list of exogenous factors that all have an influence on any proposed figure of merit:

- end device spatial density;
- end device duty cycle;
- end device transmission modulation type;
- environment (rural, urban, industrial...);
- level of interference.

The sampling infrastructure possesses some levers that can be activated to achieve satisfactory performance:

- CR spatial density;
- CR sampling rate;
- CR sampling bandwidth;
- FC computing power.

There is no optimal setup and there are a lot of trade-offs between the different mentioned factors (for example, is it better to have many CRs sampling at low rates or few CRs sampling at high rates?). We will study some trade-offs throughout this thesis, such as the one between high infrastructure performance and low sampling rates. However, these are only scaling factors, and underlying mechanisms at play in the proposed infrastructure will be the main focus of our study.

0.3 Plan of the thesis and contributions

Following this general introduction, **Chapter 1** provides an extensive review of two tightly inter-connected fields: Compressive Sampling (CS) and Spectrum Sensing (SS). Fundamentally, CS theory relies on solving underdetermined systems of linear equations (USLE) (this step is also called signal reconstruction or recovery). We therefore explore conditions for good reconstruction as well as methods to solve a USLE. CS theory is complemented by several sub-Nyquist sampling schemes. Typically, to each sampling scheme corresponds a different USLE. We present the principal sampling schemes and their characteristics, with a particular attention brought to MRS. Finally, USLEs are solved using the additional assumption that the signal is sparse in some basis. In particular, knowledge of the spectral support allows the reduction of the space of possible solutions of the USLE subjacent to the MRS. We go through the principal SS techniques present in the literature, as they are a prior to signal support detection.

Chapter 2 is dedicated to novel SS techniques that will help us in our MRS-based sampling infrastructure. We start by presenting our novel K-means-based Noise Variance Estimator (KNVE), an algorithm based on K-means clustering to estimate the noise variance in frequency-sparse signals corrupted by white Gaussian noise of unknown variance. Then, we shed some light on signal support detection for sparse signals and introduce our proposed Support Estimator (SE), which estimates the signal support from samples obtained in an MRS branch, using sample-wise energy detection. Both presented methods are compatible with the MRS, though not limited to this particular sampling scheme. As the different MRS branches collect samples at different rates, the signal support estimates vary from one branch to another because of aliasing: we say that each branch partially estimates the signal support. We discuss strategies of combination of these signal support partial estimates into a global signal support estimation.

Though MRS was introduced over a decade ago, we believe it has not been studied thoroughly. In particular, the choice of the sampling rates has received little attention. To fill this gap, in **Chapter 3**, we introduce new theoretical results. Our first contribution of this chapter is a necessary condition for satisfactory resolution of the USLE related to the MRS. This condition states that the rank of the matrix representation of the USLE be as high as possible. Our second theoretical contribution is an upper-bound for the rank of this matrix, based solely on the ADC sampling rates and the number of ADCs. We demonstrate that choosing pairwise coprime sampling rates allows to reach this upper-bound. In addition, we provide simulations results underlining the relevance of the rank indicator for asserting the performance of an MRS system.

Now that these important aspects of MRS have been developed and clarified, in **Chapter 4** we present a demonstrator for a complete mono-site[4] MRS sampling infrastructure. In this processing chain, we generate narrowband (NB) signals using typical IoT characteristics (modulations, message sizes, duty cycles...) and the aggregated wideband (sparse) signal undergoes propagation through a Rayleigh channel and an AWGN channel of unknown variance. We sample this signal, estimate the noise variance and the signal frequency-domain support, and reconstruct the original NB signals transmitted by the devices using several solvers. Finally, we evaluate the performance of our system for different sets of parameters, using as a metric the root mean square error (RMSE) between the received noisy signal and the undersampled-then-reconstructed signal. In the light of the simulation results, we provide elements to help scale the sampling infrastructure. Finally, we present ways to make the sampling infrastructure adaptive to changes in radio environment.

The infrastructure studied in Chapter 4 is mono-site. This is an intermediary step towards a multi-site infrastructure, such as the one envisioned in Section 0.2.3 of this general introduction. In the general conclusion, we discuss the opportunities as well as the challenges associated

[4] A mono-site sampling infrastructure is an infrastructure equipped with one physical reception antenna, contrary to a multi-site infrastructure in which several antennas are located at distinct geographical sites.

with a multi-site sampling infrastructure based on Spatial Diversity MRS (SD-MRS). We also explore various possible improvements and refinements to the sampling infrastructure proposed in Chapter 4.

Chapter 1

Review of Compressed Sensing and Spectrum Sensing

1.1 Introduction

In this chapter, we will first introduce a self-contained overview of the Compressed Sensing (CS) framework. Because an underlying structure is present in many signals of interest, including telecommunications signals, the information conveyed by such signals can be represented by fewer coefficients than in traditional signal representations. This assertion lies at the heart of the CS framework. In telecommunications, an example of a traditional signal representation is the temporal signal sampled uniformly at the Nyquist rate. In practice, to each hardware implementation of CS corresponds a sampling matrix in $\mathbb{K}^{m\times n}$, where $\mathbb{K} \in \{\mathbb{C}, \mathbb{R}\}$. Because a CS scheme takes fewer samples than traditional Nyquist sampling, we have $m < n$; thus, the sampling matrix is equivalent to an Underdetermined System of Linear Equations (USLE), which has either zero or an infinite number of solutions. Assumptions on the structure of the signal then help find the solution closest to the original signal. Section 1.2 will consider the main theoretical aspects behind CS, while several applications and hardware architectures of CS in telecommunications will be described in Section 1.3.

A concept that somewhat intersects with CS is Spectrum Sensing (SS). SS is a core component of the Cognitive Radio (CR) paradigm. It consists in monitoring spectrum usage and exploiting this knowledge for a variety of applications. Its principal application is the detection of Primary Users (PUs) for Dynamic Spectrum Access (DSA). As mentioned in Section 0.2.2 of the General Introduction, CS is considered as an enabler for wideband SS (WB-SS) in the DSA use case. However, by giving information about spectrum usage, SS can also be used to provide knowledge about the underlying structure of the signal, in order to solve a given USLE obtained through CS. In Section 1.4, we will present two flavors of SS: narrowband- and wideband-SS. Narrowband-SS (NB-SS) permits signal detection in a given narrow band, which is crucial to DSA because it helps determine whether or not the spectrum is occupied by another user. WB-SS makes signal support estimation possible (we consider in this regard that the signal is composed of several narrowband components), which is useful for solving a

CS-originated USLE for the purpose of signal recovery.

1.2 Theoretical aspects of Compressed Sensing

1.2.1 Preliminary considerations on signal and vector spaces

1.2.1.1 Continuous or discrete signals?

Consider a real-valued[1] continuous baseband signal $x(t)$, whose support (set of frequencies where x is non-zero) is comprised within $[-f_{\max}, f_{\max}]$. Although a strictly band-limited signal should be of infinite duration, we suppose $x(t)$ is of finite duration Δ. The continuous signal $x(t)$ is sampled at the Nyquist rate $f_{\text{Nyq}} = 2f_{\max}$. Let $x = [x[0] \dots x[n-1]]^T, n = \Delta f_{\text{Nyq}}$ be the vector of discrete time-domain representations (samples) of $x(t)$. Each of its values is obtained as follows:

$$x[k] = x(\frac{k}{f_{\text{Nyq}}}), 0 \leq k \leq n-1. \tag{1.1}$$

Using the Whittaker-Shannon interpolation formulas, the continuous signal $x(t)$ can be completely recovered from the Nyquist-rate sampled signal $x[k]$ (the interpolation error obtained as a consequence of the finiteness of $x(t)$ is not taken into account). For this reason, throughout the remainder of this manuscript, "signal recovery" (or its synonym "signal reconstruction") will refer to the recovery of the finite-length discrete signal $x[k]$ and not that of its continuous counterpart $x(t)$. This discrete, finite-length signal $x[k]$ (also referred to as x) is an element of $\mathbb{R}^n$.

1.2.1.2 Norms

A norm on vector space V is a non-negative linear application $||\cdot|| : V \longrightarrow \mathbb{R}$ verifying three properties. For all $a, b \in V$ and $\lambda \in \mathbb{R}$:

- $||a|| = 0 \leftrightarrow a = 0$ (a norm is *positive definite*);
- $||\lambda a|| = |\lambda|||a||$ (a norm is *absolutely homogeneous*);
- $||a + b|| \leq ||a|| + ||b||$ (a norm verifies the *triangular inequality*).

Different norms are useful to represent distinct physical phenomena. We now introduce several norms on vector space $\mathbb{R}^n$ and the quantities they physically represent.

For a vector $x = [x_1 \dots x_n] \in \mathbb{R}^n$, the ℓ_p-norm $||\cdot||_p$ $(1 \leq p < +\infty)$ is defined as

$$||x||_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}. \tag{1.2}$$

Further, the ℓ_∞-norm $||\cdot||_\infty$ is defined as

$$||x||_\infty = \max_{1 \leq i \leq n} |x_i|. \tag{1.3}$$

[1]Throughout this document, almost all considered signals are complex. However, unless mentioned otherwise, the real and imaginary parts of these signals are processed separately as real-valued signals and summed afterwards.

Let the support of x be the set of indices of nonzeros of x:

$$\text{supp}(x) = \{i, x_i \neq 0\}. \tag{1.4}$$

The ℓ_0-pseudonorm $||\cdot||_0$ is defined as

$$||x||_0 = \text{Card}(\text{supp}(x)), \tag{1.5}$$

where $\text{Card}(\cdot)$ denotes the number of elements in a set. Because it is not absolutely homogeneous, $||\cdot||_0$ is not a norm, but rather a pseudonorm.

Examples of norms representing different physical phenomena are the ℓ_2-norm and the ℓ_0-pseudonorm. The ℓ_2-norm typically represents the square root of the energy in a signal x. The ℓ_0-pseudonorm is a measure of sparsity of a signal, that is, of how much a signal's information is contained within a few coefficients, a concept that is key to CS as we will see in Section 1.2.2.

1.2.1.3 Bases

A set $\{\phi_i\}_{i=1}^n$ of elements of $\mathbb{R}^n$ is said to be a basis of $\mathbb{R}^n$ if it consists of linearly independent vectors. For any $x \in \mathbb{R}^n$, there is a unique set of coefficients $\{c_i\}_{i=1}^n$ that describes x in the basis:

$$x = \sum_{i=1}^{n} c_i \phi_i = \Phi c, \tag{1.6}$$

where Φ is the $n \times n$ matrix with columns given by ϕ_i and c is the vector of the c_i values.

If the basis is orthonormal, meaning

$$\langle \phi_i, \phi_j \rangle = \begin{cases} 1, i = j \\ 0, i \neq j \end{cases}, \tag{1.7}$$

we have $\Phi^T \Phi = I_n$ where I_n is the $n \times n$ identity matrix. c can then be easily obtained:

$$c = \Phi^T x. \tag{1.8}$$

1.2.1.4 Frames

A frame is a set $\{\phi_i\}_{i=1}^n$ of elements of $\mathbb{R}^d, d < n$ of associated matrix $\Phi \in \mathbb{R}^{d \times n}$ that verifies, for any $x \in \mathbb{R}^d$:

$$A||x||_2^2 \leq ||\Phi^T x||_2^2 \leq B||x||_2^2, \tag{1.9}$$

where $0 < A \leq B < \infty$.

A frame is a generalization of a basis where the number of elements in the set $\{\phi_i\}$ is higher than the dimension of the considered space. Due to the linear dependence of the ϕ_i in a frame, for any given x, there are an infinity of vectors c that verify $x = \Phi c$.

One way to obtain feasible coefficients is to use the dual frame $\tilde{\Phi}$. A dual frame verifies

$$\Phi \tilde{\Phi}^T = \tilde{\Phi} \Phi^T = I_n. \tag{1.10}$$

Similarly to (1.8), a dual frame allows to obtain c from any x:

$$c = \tilde{\Phi}^T x. \tag{1.11}$$

A particular dual frame is the canonical dual frame, also known as the Moore-Penrose pseudoinverse (MPPI), defined by

$$\tilde{\Phi} = (\Phi\Phi^T)^{-1}\Phi. \tag{1.12}$$

Note that $\Phi\Phi^T$ is invertible if and only if the rows of Φ are linearly independent.

The MPPI solves the "least-squares" problem, meaning that the coefficients c_{mppi} obtained by choosing the MPPI as the dual frame in (1.11) verifies

$$||c_{\text{mppi}}||_2^2 \leq ||c||_2^2 \text{ subject to } x = \Phi c. \tag{1.13}$$

Bases and frames are routinely called dictionaries and their elements are called atoms.

1.2.2 Signal sparsity

Some signals can be represented – or approximated – by a linear combination of only a few atoms of a dictionary, meaning that the majority of atoms in the dictionary contain little to no information about the signal.

Consider the aforementioned signal x. x is said to be k-sparse if it has k or fewer nonzero values, meaning that $||x||_0 \leq k$. Let

$$\Sigma_k = \{x, ||x||_0 \leq k\} \tag{1.14}$$

be the set of all k-sparse vectors. Other measures of sparsity have been proposed and compared in [29]. Note that Σ_k is not closed under addition and that for $x_1, x_2 \in \Sigma_k$, we have $x_1 + x_2 \in \Sigma_{2k}$.

A sparse signal is a signal consisting mostly of zeros, that is, $k \ll n$. Often, a signal[2] might not be directly sparse, but its representation in another basis Φ is. Consider non-sparse signal s. If there is a basis Φ so that $x = \Phi s$ with $||x||_0 \leq k$, we will consider that x is k-sparse, and that s is k-sparse *in some basis*. Φ is called the sparsifying basis. Fig. 1.1 displays the sparsifying transform, where a non-sparse signal of size $N = 10$ samples becomes a k-sparse signal x with $k = 3$.

Signal sparsity is key to data compression [30]: by finding the right dictionary in which to express the signals, most coefficients describing the signal can be dropped out, achieving a reduction in signal size. The compression can be either lossless (no information at all is lost) or lossy (some information is lost during compression). A pioneering work in lossy compression was the conception of the Discrete Cosine Transform (DCT) by Ahmed et al [31]. This transform concentrates most of the signal information in a few low-frequency components, making it possible to drop most higher-frequency components and thus reduce the size of the signal. This transform is now a central part to many compression and coding standards, such as JPEG, H.265 and MP3. Note that a lot of signals, such as natural images or audio recordings, are not exactly sparse, but are approximately sparse. This means that they can be approximated

[2] More precisely, its sample representation resulting from some physical acquisition process.

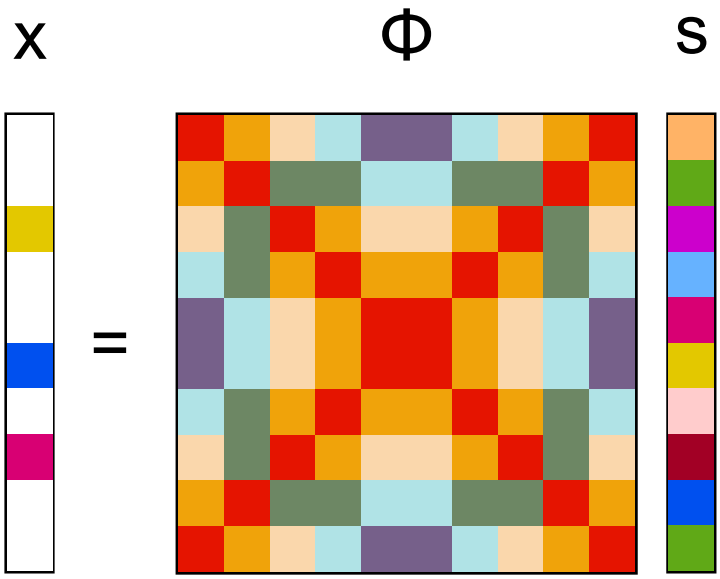


Figure 1.1: Sparsifying transform: transforming a non-sparse signal s into a k-sparse signal x through multiplication by a sparsifying basis Φ.

by sparse signals by setting the numerous small coefficients to zero: setting the $n-k$ lowest coefficients to zero yields a k-sparse approximation of the signal.

Signal sparsity is also exploited in denoising [32] and image processing [33]. Other applications of sparsity include statistics and learning theory, where sparsity has been used to avoid overfitting [34], and the study of the human visual system [35].

The difference between data compression and CS is the following. In data compression, we start with a known non-compressed signal, to which we apply an adequate transform, and this yields a sparse representation. When sampling, this strategy is suboptimal, because we would first have to sample the signal at a high, Nyquist rate, only to "throw away" the majority of coefficients of the signal at compression because they contain little information. In his groundlaying work [19], Donoho advocates that if a signal $x \in \mathbb{R}^n$ is compressible through a known transform, the compressed version of x can be obtained without going through the energy-intensive, high-rate sampling step: the required number of linear mesurements m can be much smaller than n. In another major work, Candès, Romberg and Tao [18] observe a connection between the number of necessary observations and the number of nonzeros in x.

In addition to signal sparsity, there are other models to describe signals that have the ability to be represented with far fewer atoms of a given dictionary than that of another. Union of Subspaces [36] and Finite Rate of Innovation [37] are such frameworks. In particular, they are helpful with applying CS theory to analog, non-discretized signals for which (1.14) is not relevant. Though these models will not be studied in the remainder of this document, the interested reader is referred to Chapters 3 and 4 of [38].

1.2.3 Sampling matrix properties

CS consists in acquiring a signal $x \in \mathbb{R}^n$ through $m < n$ linear measurements:

$$y = Ax, \tag{1.15}$$

where $y \in \mathbb{R}^m$ is called the measurement vector and $A \in \mathbb{R}^{m \times n}$ is called the sampling matrix. Some applications, such as DSA, do not require the recovery of x, but rather *characteristics* derived from x; however, for completeness, we will aim to recover x from (1.15). Fig. 1.2 represents the measurement acquisition process of CS.

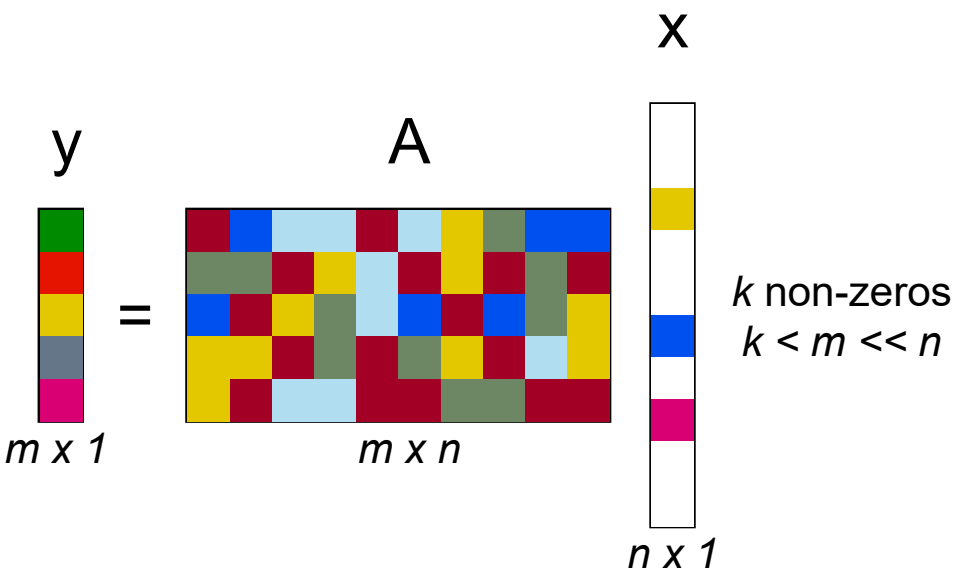


Figure 1.2: Compressed Sensing: collecting $m \ll n$ measurements through a sampling matrix A. The m-sized vector y contains all the information present in n-sized x and represents it in fewer coefficients.

Since A has more columns than rows, it corresponds to a USLE. As a consequence, with y and A known, there are either zero or an infinity of solutions for x to (1.15). Since x is supposed to be sparse, we will try to recover the *sparsest* $\hat{x}$ that verifies $y = A\hat{x}$. In other words, we will consider the ℓ_0-pseudonorm minimization problem:

$$\hat{x} = \text{argmin}_z ||z||_0 \text{ subject to } y = Az. \tag{1.16}$$

The processing of CS-acquired signals is summarized in Fig. 1.3.

An important factor of success for reco[3]very of x lies in the design of A. First, A needs to be designed so that if x is sparse, all the information in x is present in y. Second, the design of A should allow for a pratical and effective recovery of x from the measurement vector y.

We will now present several sampling matrix properties that provide guarantees for signal recovery, and see how these properties influence the sampling matrix design procedure.

[3]Throughout this document, almost all considered signals are complex. However, unless mentioned otherwise, the real and imaginary parts of these signals are processed separately as real-valued signals and summed afterwards.

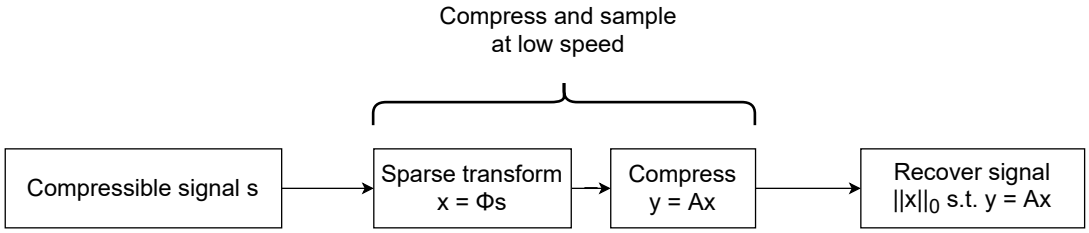


Figure 1.3: Processing CS-acquired signals.

1.2.3.1 Spark

Consider distinct vectors $x, x' \in \Sigma_k$ for which $Ax = Ax'$. This situation is not desirable because it is impossible to recover the actual original vector x from measurement vector y. In this case, notice we have both $x - x' \in \Sigma_{2k}$ and $x - x'$ in the nullspace of A. For A to represent an injective function of Σ_k into $\mathbb{R}^m$, there should be no vector both in $\ker A \setminus \{0\}$ and Σ_{2k}. The spark introduced in [39] helps us characterize A in this regard.

Definition 1.1 (Spark)**.** *The spark of a given matrix A is the smallest number j of columns of A for which a subset of j columns of A are linearly dependent.*

The following holds:

Theorem 1.2. *If it exists, a solution $x \in \Sigma_k$ to $y = Ax$ (for any $y \in \mathbb{R}^m$) is unique if and only if spark$(A) > 2k$.*

Proof is given in [39].

From the definition of the spark, we can see that spark$(A) \in [2, m+1]$. As a consequence of Theorem 1.2, the minimum number of measurements required is $m \geq 2k$. This is an elegant rewriting of the Landau bound [21]. Note that if the original signal is not sparse at all ($k = n$), then the spark condition yields the Nyquist rate.

1.2.3.2 Null Space Property

In general, the ℓ_0-pseudonorm minimization problem defined in (1.16) is NP-hard [40]. As an alternative, the technique of ℓ_1-relaxation is commonly used. It consists in solving the ℓ_1-problem:

$$\hat{x} = \text{argmin}_z ||z||_1 \text{ subject to } y = Az, \tag{1.17}$$

instead of the ℓ_0 problem. The advantage of this relaxation is that the ℓ_1-problem is convex, and therefore solvable through linear programming techniques. However, for the l_1-relaxation to be relevant, it is necessary that the ℓ_0- and ℓ_1-problems have the same answer $\hat{x}$. The null space property (NSP) is one way to guarantee equivalence:

Definition 1.3 (Null space property)**.** *A matrix A satisfies the NSP of order s if for all index sets S with $s = Card(S) \leq n$, we have $||\eta_S||_1 < ||\eta_{S^C}||_1$ for every $\eta \in \ker A \setminus \{0\}$.*

The NSP allows the following recovery condition:

Theorem 1.4. *Let $A \in \mathbb{R}^{m \times n}$. Then every k-sparse signal x is the unique solution to the ℓ_1-relaxation problem with $y = Ax$ if and only if A satisfies the NSP of order k.*

Proof is given in [41].

1.2.3.3 Restricted Isometry Property

While the NSP is suitable for noiseless measurements, error quantization or noise contamination can prevent a matrix A satisfying the NSP from successfully recovering x. To account for impairments during the acquisition of measurements, the Restricted Isometry Property (RIP) was introduced by Candès and Tao in [42] and has since then been one of the most used matrix properties in CS.

Definition 1.5 (Restricted Isometry Property)**.** *A matrix A satisfies the Restricted Isometry Property of order k if there exists a $\delta_k \in (0, 1)$ such that for all $x \in \Sigma_k$:*

$$(1 - \delta_k)||x||_2^2 \leq ||Ax||_2^2 \leq (1 + \delta_k)||x||_2^2. \tag{1.18}$$

The RIP is a powerful property because it provides stability for k-sparse vectors. Indeed, the distance between any given pair of k-sparse vectors is approximately preserved by a matrix satisfying the RIP of order $2k$. This means that a matrix satisfying the RIP corresponds to a well-posed problem and solutions will be numerically stable: this is particularly important if the input signal is impaired by noise or quantization errors.

Another way to consider the RIP is by noticing that a matrix satisfying the RIP of order $2k$ is almost orthonormal when operating on vectors in Σ_k.

It is possible to propose bounds for the minimum number of measurements m needed for the matrix A to satisfy the RIP of order k. Davenport [43] proposes the following lower bound, for given n and k:

Theorem 1.6. *Let A an $m \times n$ matrix that satisfies the RIP of order $2k$ with constant $\delta_{2k} \in (0, 1)$. Then*

$$m \geq C_\delta k \log\left(\frac{n}{k}\right), \tag{1.19}$$

where $C_\delta < 1$ depends only on δ_{2k}.

Other similar bounds are proposed in [38].

Finally, the RIP is related to the NSP, and is even strictly stronger than the NSP: if a matrix satisfies the RIP, then it also satisfies the NSP. Proof of this can be found in [38].

1.2.3.4 Coherence

The spark, NSP and RIP all propose guarantees for the recovery of k-sparse signals, sometimes with stability and robustness to impairments. However, verifying that a matrix A satisfies any of these properties usually involves a combinatorial search over $\binom{n}{k}$ submatrices. The coherence of a matrix [44] is a property that measures the correlation between the distinct columns of A. It is easy to compute and provides guarantees for signal recovery.

Definition 1.7 (Coherence)**.** *The coherence $\mu(A)$ of a matrix A is the largest absolute inner product between any two distinct columns a_i, a_j of A:*

$$\mu(A) = \max_{1 \leq i,j \leq n, i \neq j} \frac{|\langle a_i, a_j \rangle|}{||a_i||_2 ||a_j||_2}. \tag{1.20}$$

The coherence is upper-bounded by 1 and lower-bounded by the Welch bound [45]:

$$\sqrt{\frac{n-m}{m(n-1)}} \leq \mu(A) \leq 1. \tag{1.21}$$

In particular, if $m \ll n$, the lower bound can be approximated by $\mu(A) \geq 1/\sqrt{m}$.

The coherence of a matrix can be linked to its spark by the following relation.

Lemma 1.8. *For any matrix $A \in \mathbb{R}^{m \times n}$,*

$$\textit{spark}(A) \geq 1 + \frac{1}{\mu(A)}. \tag{1.22}$$

A proof of this lemma using the Gram matrix $G = A^T A$ can be found in [39].

Combining Theorem 1.2 and Lemma 1.8 yields a direct relation between the matrix coherence and the required level of sparsity for uniqueness of recovery:

Theorem 1.9. *If*

$$k < \frac{1}{2}\left(1 + \frac{1}{\mu(A)}\right), \tag{1.23}$$

then at most one signal $x \in \Sigma_k$ verifies $y = Ax$ for any given measurement vector $y \in \mathbb{R}^m$.

From the Welch bound (1.21) and Theorem 1.9, we can exhibit a relation between sparsity order k and number of measurements m guaranteeing uniqueness using the coherence property:

$$k = \mathcal{O}(\sqrt{m}). \tag{1.24}$$

Notice that for a given level of sparsity k, a matrix A typically requires more measurements m to satisfy the coherence property than the RIP: in the order of $k \log\left(\frac{n}{k}\right)$ for the RIP (1.19) and k^2 for coherence. Remember that these values are lower bounds.

1.2.4 Sampling matrix construction

Now that we have properties to ascertain that a given matrix A will be suitable for uniquely and exactly recovering an original k-sparse signal x from a measurement vector y of length m, we wish to effectively construct sampling matrices that satisfy the aforementioned properties. Two approaches have drawn particular attention for this purpose: deterministic and random matrices.

1.2.4.1 Deterministic matrices

A $m \times n$ Vandermonde matrix V built with m distinct scalars satisfies $\text{spark}(V) = m + 1$. However, when n is large, V is poorly conditioned, making the recovery of $x \in \Sigma_k$ from $y = Ax$ numerically unstable [46].

A construction based on the Gabor frame obtained from a sequence known as the Alltop sequence creates a $m \times m^2$ sampling matrix that reaches the lower bound coherence $\mu(A) = 1/\sqrt{m}$ [47].

RIP-satisfying deterministic matrices have also been an ongoing subject of research for years ([48, 49, 50, 51, 52]). However, the proposed constructions usually require a large number of measurements m (for example, in the order of magnitude of $m = \mathcal{O}(k^2 \log n)$ in [49]), and most known approaches are confronted with the so-called $\sqrt{m}$ bottleneck, meaning that deterministic RIP-satisfying $m \times n$ matrices can only deal with signals x of sparsity level $\sqrt{m}$. Bourgain et al. [48] provided a way to go beyond this bottleneck using the *flat RIP*. Their theoretical contribution was proposed alongside a construction of a RIP-satisfying matrix that allows a sheer gain over the $\sqrt{m}$ bottleneck. In [50], Bandeira et al. use the theoretical contribution of Bourgain et al. [48] to conceive a method that can demonstrate RIP for levels of sparsity $k > \sqrt{m}$. Nonetheless, due to high required numbers of measurements, deterministic-constructed, RIP-satisfying matrices are typically inadequate for real-world sampling environments.

1.2.4.2 Random matrices

Random matrices have also stimulated a lot of interest because they can satisfy the RIP with fewer measurements m and/or lower values of δ_k with high probabilities.

A major result states that a random matrix has a high probability of satisfying the RIP if its entries follow any sub-gaussian distribution. In particular, Theorem 5.65 of [38] states the following:

Theorem 1.10. *If a $m \times n$ matrix A with*

$$m = \mathcal{O}(k \log(n/k)/\delta_{2k}^2) \tag{1.25}$$

has its entries chosen from a sub-gaussian distribution, then A satisfies the RIP of order $2k$ with a probability of at least

$$p \geq 1 - 2e^{-c\delta_{2k}^2 m}, \tag{1.26}$$

with c a positive constant.

Through this result, the optimal number of measurements presented in (1.19) is reached up to a constant, which makes a strong case for random matrices.

Another advantage of random matrices is that they allow for measurements that are *democratic* [53], that is, each measurement contains approximately the same quantity of information. It also means that any sufficiently large subset of measurements can effectively be used for signal recovery. For this reason, random matrix constructions bring about robustness by redundancy: taking a few more measurements allows to "lose" some measurements to corruption or channel impairments.

One additional consideration is that a signal x might only be k-sparse in respect to some basis Φ. In this scenario, the matrix that should satisfy the RIP is no longer A, but $A\Phi$. If A is built in a deterministic fashion, construction should take Φ into account. On the contrary, if

A is a random matrix, $A\Phi$ will be very likely to satisfy the RIP with the same probability as A[4], meaning that it is unecessary to take Φ into consideration for the construction of A. As a consequence, random matrices have a clear advantage over deterministic sampling matrices.

The major caveat of the random matrix approach is that it seems impractical to build in an actual hardware setup. A way to mitigate this problem consists in allowing for some randomness to be implemented in hardware architectures (which will be covered in a further section). These architectures correspond to sampling matrices A that exhibit more structure than fully-random matrices. However, these more structured matrices can, in some cases, satisfy the RIP or have a low coherence.

1.2.5 Beyond properties of guarantee of recovery

The different properties presented in Section 1.2.3 and the guarantees they provide regarding recovery only describe one side of the picture. Because the properties are sufficient and not necessary, for a given matrix A, there is a gap between the recovery guarantees and the actual recovery possibilities. For example, in a simple experiment, Bruckstein et al. [55] were able to exhibit a case where the actual signal recovery algorithm (usual signal recovery methods will be covered in Section 1.2.6) could on average successfully recover a signal with a level of sparsity k 26 times higher than what coherence guaranteed.

1.2.5.1 The gap between guaranteed performance and actual recovery performance

Beyond this simple yet eloquent example, there is a documented gap [56, 57, 55] between what a satisfied property (RIP, low coherence, spark, and others) guarantees and what is achievable. One reason that could explain this gap is that a property has to guarantee k-sparse signal recovery for the *worst-case* scenario, which might correspond to an extremely limited subset of all possible signals $x \in \Sigma_k$ and might occur with a very low (if not zero) probability. On the contrary, the *typical behavior* of the system, seen as the combination of the original signal x, the sampling matrix A and the recovery method, does not account for the worst-case scenario and can incur satisfactory recovery of k'-sparse signals ($k' > k$) in most cases.

For the coherence property, the worst-case scenario unravels in the following fashion. Recall that if any two columns of a sampling matrix A have a high inner product, the coherence of the *whole* matrix will be too high to guarantee signal recovery for any substantial sparsity level k. However, the matrix may have thousands of columns, and the two aforementioned correlated columns will only project two elements of x (out of thousands) onto subspaces close to each other. In the extreme example that two columns a_i and a_j of A are identical, the coherence will be highest at $\mu(A) = 1$, only guaranteeing the recovery of a 1-sparse signal. Consider a 2-sparse signal ξ. In the worst-case scenario, the two nonzeros of ξ are located at the ith and jth indices, thus jeopardizing recovery of ξ. However, in the typical behavior scenario, there is a probability of $\frac{998}{1000} \times \frac{997}{999} \approx .996$ that ξ has both of its nonzeros at indices other than i or j, and

[4]This is true for matrices built with sub-gaussian distributions [54].

that recovery can be performed correctly (notwithstanding other impairments in the sampling matrix).

While there seems to be a wider gap between what a low coherence guarantees and actual signal recovery than for the other properties, the same reasoning applies to all of them. This means that signal recovery guarantees often cannot provide more than a lower-bound estimate of the achievable recovery performance in the typical behavior regime.

1.2.5.2 Phase transitions

In the context of signal recovery, let us define a "success" outcome as either a perfect recovery of a sparse signal ($\hat{x} = x$) or a low-mean square error (MSE) recovery ($||\hat{x} - x||_2^2 \leq \epsilon$ for a given error $\epsilon > 0$), and a "failure" outcome as a non-successful outcome. Let us use the notations of [55] where the occupancy ratio is defined as $\delta = k/m$ and the undersampling ratio is defined as $\rho = m/n$.

Interestingly, success rate and failure rate for given sampling matrices and recovery algorithms and given values of δ and ρ are far from random [58, 55, 59]. Figure 1.4 taken from [55] depicts the "success" and "failure" phases for signal recovery versus ρ and δ. Gaussian matrices with $n = 1600$ are used. The shaded area boundaries are well defined in both panels, and the transitions between a high success rate and a high failure rate (depicted by the red curves) are clearly visible: a small variation of k, all other parameters being equal, can cause a drop from a 100% success rate to a 0% success rate. As the number of equations n increases, the transition becomes increasingly sharp [58].

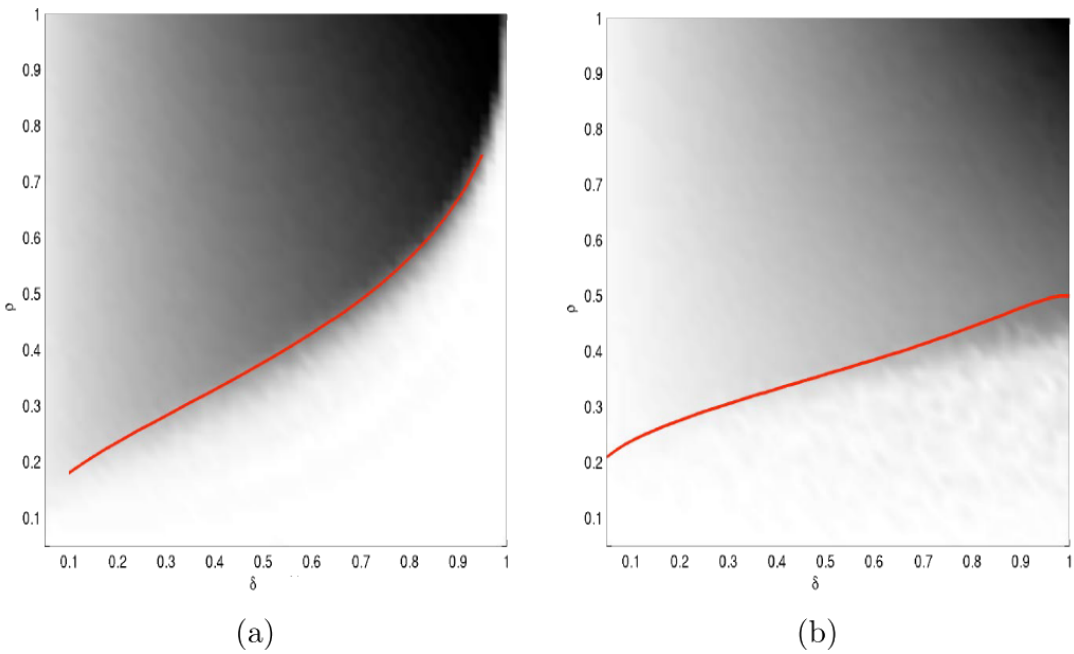


Figure 1.4: Phase transitions for two different recovery algorithms ((a): ℓ_1 minimization (b): StOMP, see Section 1.2.6). The sampling matrices are Gaussian with $n = 1600$. The different shades represent fractions of cases in which the signal recovery algorithm successfully finds the sparsest solution (darkest: 0% success rate, brightest: 100% success rate).

There is a conjectured threshold η on the degree of sparsity k of the signal in case the sampling matrix is Gaussian with $m \ll n$, and considering the recovery algorithm is ℓ_1 minimization [55]:

$$\eta = \frac{m}{2\log(n/m)}. \tag{1.27}$$

If k is – even slightly – above this threshold η, we are in the failure phase, and if k is under the threshold, we are in the success phase.

Comprehensive statistical studies of phase transitions in CS have been carried out [60, 61]. They are mostly based on Gaussian matrices and various recovery algorithms, and provide useful insights for analysis and problem resolution in CS.

1.2.6 Signal recovery

Fundamentally, recovering a signal from CS measurements consists in solving a USLE. We will now present the principal classes of algorithms for USLE resolution in the context of CS. Note that in this section, we will only cover the full recovery of original signal x. The recovered signal is noted $\hat{x}$. However, some applications, such as detection or parameter estimations, do not aim for a full signal recovery, but rather for some level of knowledge about the original signal.

1.2.6.1 ℓ_1-minimization algorithms

Perhaps the most natural approach to recovering x from measurement vector y is to solve the ℓ_0-minimization problem (1.16). However, this non-convex problem is NP-hard. Furthermore, for a general matrix A, finding an approximation of the true minimum is NP-hard [62].

To obtain a more tractable problem, a ℓ_1-relaxation has been applied to the problem to yield the ℓ_1-minimization problem (1.17). This problem is also called Basis Pursuit (BP).

Regularization has been proposed to counter BP's tendency to favor exactitude of recovery over sparsity, even in a noisy setting. A regularized problem derived from BP, called Basis Pursuit Denoising (BPDN), "ℓ_1-least squares" (L1LS) or Least Absolute Shrinkage and Selection Operator (LASSO), has been studied extensively:

$$\hat{x} = \text{argmin}_z \frac{1}{2}||Az - y||_2^2 + \lambda||z||_1, \tag{1.28}$$

where the regularization parameter $\lambda > 0$ aims at providing a balance between sparsity (brought by a low ℓ_1-norm) and recovery fidelity (brought by a low ℓ_2-norm).

Several approaches for determining the optimal λ are proposed in [63, 64].

Note that for any given value of λ, solving the unconstrained formulation of BPDN (1.28) will yield the same result as solving the constrained version of the problem:

$$\hat{x} = \text{argmin}_z ||z||_1 \text{ subject to } ||Az - y||_2 \leq \epsilon, \tag{1.29}$$

although the relation between ϵ and λ is unknown *a priori*. An interest of the constrained formulation of BPDN lies in its similitude to a natural parameterization of the problem, where solution RMSE $||A\hat{x} - y||_2$ is bounded by a noise or quantization error threshold.

Since BP and BPDN are convex, they can be solved by any given general-purpose convex optimization software using methods such as Dantzig's simplex algorithm [65]. However, several algorithms have been proposed to solve the ℓ_1-minimization problem specifically in the context of CS. These algorithms are based on various convex optimization techniques: gradient descent for the (Fast) Iterative Soft-Thresholding Algorithm (FISTA) [66], gradient projections [67], coordinate descent [68], fixed-point continuation [69], predual resolution for the Predual Proximal Point Algorithm (PPPA) [70], specialized interior-point method [71] to name a few.

Though ℓ_1-minimization algorithms provide satisfying results in terms of MSE between original signal x and solution $\hat{x}$, one of their drawbacks is that they typically involve heavy computations and tend to be slow for larger problems.

1.2.6.2 Greedy algorithms

The family of greedy algorithms has garnered a lot of attention for sparse signal recovery from compressed measurements. Greedy algorithms obtain solutions through iterative approximations of the signal coefficients or signal support and run until a convergence criterion is met. Each iteration can involve a first-order convex optimization technique, such as a gradient descent. Performance of greedy algorithms routinely match that of ℓ_1-minimization algorithms [38], with smaller complexities and lower computation times. We will now review two important greedy algorithms and their variants.

Orthogonal Matching Pursuit A very popular greedy algorithm is Orthogonal Matching Pursuit (OMP) [44]. The principle of OMP is to iteratively pick the columns of A most strongly correlated to the remaining part of the measurement vector y. Then, the contribution of the selected column of A to the mesurements is removed and the algorithm iterates on the residual. At each step i, the level of sparsity of the iteratively-built solution $||\hat{x}_i||_0$ increases by 1. The algorithm stops after a predefined number of steps that can be set to be the estimated number of nonzeros of x (though other stopping criteria, such as an ℓ_2 error condition can be used as well). A more formal description of OMP is presented in Algorithm 1.1 (formulation originally taken from [38]). In Algorithm 1.1, † denotes the Moore-Penrose pseudo inverse (MPPI) of a matrix, $\cdot^C$ denotes the complement and H_k represents the *hard thresholding operator*, which selects the k elements of x with the highest magnitude and set all other entries to zero.

OMP has some empirical guarantees on signal recovery [44], with a required number of measurements of roughly $m = \mathcal{O}(Ck \log n)$ for a Gaussian sampling matrix A (with C a positive constant). Complexity depends on the practical implementation of OMP. While there is a trade-off between time and memory complexities, time complexity can be as low as $\mathcal{O}(nk + mk)$ [72].

Extensions of OMP have been proposed, that provide gains in speed, efficiency, or stability. Block OMP (BOMP, [73]) yields better recovery than OMP for block-sparse signals. Stagewise OMP (StOMP, [59]) enhances OMP by using hard thresholding with a specific threshold in order to put several residual entries in S_i at Step 2 of Algorithm 1.1's for loop. Similarly, Regularized OMP (ROMP, [74]) updates S_i with not one but several entries of the residual, based on the similarity of their magnitudes. Compressive Sampling Matching Pursuit (CoSaMP, [75])

Algorithm 1.1: Orthogonal Matching Pursuit (OMP)

Input: Sampling matrix A, measurement vector y, estimate of number of nonzeros $\hat{k}$

Initialize: $\hat{x}_0 = 0, r_0 = y, S_0 = \{\}$

for $i = 1$; $i = i + 1$; $i < \hat{k}$ **do**

- $g_i \leftarrow A^T r_{i-1}$ (form signal estimate from residual)
- $S_i \leftarrow S_{i-1} \cup \text{supp}(H_1(g_i))$ (update support estimate with largest residual entry)
- $A[S_i]^\dagger \leftarrow MPPI(A[S_i])$ (compute MPPI of $A[S_i]$)
- $\hat{x}_i[S_i] \leftarrow A[S_i]^\dagger y, \hat{x}_i[S_i^C] \leftarrow 0$ (update estimate)
- $r_i \leftarrow y - A\hat{x}_i$ (update residual)

end

Output: Sparse signal estimate $\hat{x}$

improves upon ROMP and includes concepts found in combinatorial algorithms to guarantee speed and to provide rigorous error bounds.

Iterative Hard Thresholding Iterative Thresholding algorithms, like Iterative Hard Thresholding (IHT) or Subspace Pursuit (SP) have also played a prominent role in the greedy algorithm literature. We now focus on IHT. Akin to projected gradient approaches, IHT iteratively builds the solution by performing a nonlinear shrinkage (hard thresholding) after a gradient descent step. The algorithm runs until a stopping criterion is met, for example an ℓ_2 error condition. IHT is described more precisely in Algorithm 1.2.

Algorithm 1.2: Iterative Hard Thresholding (IHT)

Input: Sampling matrix A, measurement vector y, estimate of number of nonzeros $\hat{k}$, step size μ

Initialize: $\hat{x}_0 = 0$

for $i = 1$; $i = i + 1$ *until stopping criterion is met* **do**

- $\hat{x}_i = H_{\hat{k}}(\hat{x}_{i-1} + \mu A^T(y - A\hat{x}_{i-1}))$

end

Output: Sparse signal estimate $\hat{x}$

IHT has recovery guarantees of the same order as convex optimization approaches, whenever the RIP holds [76]. However, while worst-case scenarios are covered with theoretical RIP-related guarantees, the comparison of numerical results [77] of typical-behavior scenarios for IHT, OMP and convex optimization algorithms seems to be at the expense of IHT.

Variations and improvements of IHT have been put forward. In order to improve numerical results without compromising on convergence or recovery guarantees, Normalized IHT [78] proposes to compute an optimal step size μ of IHT at each iteration. Using another approach, Iterative Soft Feedback Thresholding (ISFT, [79]) substitutes the hard thresholding step for a soft-feedback step. This allows to take the discrete nature of the signal into account to yield higher performance than IHT without any significant increase in complexity. Block Normalized

IHT (BNIHT, [80]) has a performance superior to IHT for signals with the additional structure feature of block sparsity.

1.2.6.3 Other approaches to signal recovery

Greedy and ℓ_1-minimization algorithms for CS have been the focus to a tremendous amount of research. However, other classes of algorithms, that sometimes emerged in domains far different from CS, can be applied to solve the same problem as CS. We now quickly mention three such classes of algorithms.

Combinatorial methods Combinatorial methods have been applied to problems that have a strong connection to CS and the resolution of USLEs with the hypothesis of signal sparsity. One of such problems is that of combinatorial group testing [81]. CS approaches using combinatorial group testing have been researched notably in the biomedical field [82, 83]. In combinatorial group testing, one wishes to separate k anomalous or defective items out of n items in total, but does not wish to test all n items for defectiveness. In this scenario, the vector x indicates the defective items, that is, $x_i \neq 0$ for the k anomalous items and $x_i = 0$ otherwise. Each test for defectiveness on a subset of items is akin to a measurement y_i in CS. If the outputs of the tests are linear with respect to the inputs, defining a binary matrix A which elements are $a_{ij} = 1$ if the jth item is part of the ith test and 0 otherwise yields an USLE $y = Ax$. Recovering the set x of k anomalous items is then the same problem as sparse signal recovery in CS.

In combinatorial group testing, the designer is often considered to have full control over the sampling matrix A. This is a significant difference from practical telecommunications applications of CS, as we will see in a later section. In order to gain speed, A is often sparse itself, leading the way for low-complexity, faster algorithms [84].

Bayesian Compressive Sensing Generally speaking, Bayesian inference consists in updating the probability of a hypothesis as more information becomes available. This very rich framework has been successfully applied to CS [85, 86]. The Bayesian viewpoint of CS is that it consists in a linear-regression problem with a prior that the original signal x is sparse. More formally, the solution to the unconstrained formulation of BPDN (1.28) corresponds to a maximum *a posteriori* estimate for x using the Laplace sparseness prior, a prior that is popular to promote sparseness on x [87, 88]. Relevance Vector Machines have also been used to solve the regression problem with the hypothesis of sparsity [85].

Improvements upon Bayesian Compressive Sensing include the use of the variational Bayesian framework for CS [89] and hierarchical Bayesian algorithms [90]. Note that [90], unlike much of the literature, applies the Bayesian framework to a sampling matrix A that is not random, but rather a custom interpolation matrix. This suggests that the Bayesian approach could be used in telecommunications applications, where random matrices are, as previously mentioned, hard to build in practice.

More material on Bayesian inference applied to CS can be found in Chapter 6 of [38].

Deep Learning The Deep Learning (DL) revolution that started to take place in the early 2010s has brought a technological rupture to dozens of various fields and applications, from image classification [91] to network traffic control [92] through natural language processing [93] and self driving [94]. Because DL excels at automatic feature recognition, it can take advantage from signal sparsity (and other structural features) to recover a signal from compressed measurements [95]. DL can also be combined with ℓ_1-minimization algorithms like (F)ISTA [96, 97] or used to directly recover the sparse signal support [97].

DL-based approaches to sparse signal recovery have shown interesting results but not the radical breakthrough experienced in other domains so far.

1.2.7 Conclusion on the theoretical aspects of CS

We have now covered the basics of CS, a powerful mathematical framework aiming at the recovery of signals from limited quantities of measurements. After witnessing that at the core of CS lies a USLE $y = Ax$ with assumptions on original signal x, we have studied the implications of the design of A on the theoretical guarantees of signal recovery, and understood that these guarantees offer a limited insight into actual recovery performance in typical behavior scenarios. We have then explored a few methods and algorithms used to perform signal recovery from the measurement vector y and sampling matrix A. For more details on sparse signal recovery, the interested reader can refer to [98, 38].

Keeping the CS mathematical framework in mind, we now turn to more practical considerations.

The CS framework is very broad and often allows for an impactful reduction of the number of measurements. In the following section, we will present some cases in which CS has been applied in telecommunications.

1.3 Implementation of Compressed Sensing in Telecommunications

Various fields[5] have taken advantage of CS and its potential to reduce the number of acquired measurements. Uses of CS emerge when two conditions are met: (i) the signal to acquire is sparse in some basis and (ii) measuring the signal is impractical, complicated or costly. These conditions are the common denominator between all the applications for which the use of CS has been proposed, although the practical details of how these conditions are met (*e.g.* source of sparsity, corresponding representation basis, and so on) depend on the considered field and application.

Many applications of CS for telecommunications have been proposed. In this section, we will first present the most prominent applications of CS in communications networks. We will then focus on the physical layer and present some CS-based implementation schemes for signal sampling.

[5]Some examples include medical tomography [99], radio astronomy [100] and photography [101].

1.3.1 Applications of Compressed Sensing in Telecommunications

CS is an elegant framework which formalizes intuitive reasonings regarding the acquisition of data. Its broad scope has permitted applications in many domains within the communications field. To help the reader grasp how broad CS can span, we present three representative applications of CS to communications networks: spectrum sensing, detection of active devices in IoT systems and channel estimation.

1.3.1.1 Cognitive Radio: Spectrum Sensing for Dynamic Spectrum Access

The Cognitive Radio (CR) technology relies on becoming aware of the surrounding radio environment for real-time adjustment of transmission parameters. Envisioned as a response to the problem of the inefficient usage of limited spectrum, CR can be used for the purpose of finding available communication holes in the spectrum. Spectrum Sensing (SS) is a key component of CR because it enables a device to dynamically scan the surrounding radio environment. A more detailed introduction to CR and SS will be provided in Section 1.4 of this chapter. For now, we will see how SS can benefit from CS techniques.

In order to find available communication holes, a CR system should be able to scan a wideband of frequencies, up to a few GHz for example. However, directly sampling such a wideband would be both impractical and inefficient. First, it would be impractical, as it would require energy-intensive, high-speed ADCs. This requirement would face hardware limitations on ADCs [17]. Furthermore, the high volume of samples would necessitate powerful and power-hungry data processing infrastructure. Second, it would be inefficient: measurement campaigns have shown that many locations of the spectrum are underutilized, meaning that at any given time, a wideband signal covering several GHz of bandwidth typically exhibits sparsity.

An option is to subdivide a wideband into adjacent narrowbands, and to scan these narrowbands one-by-one, using conventional techniques, in order to find transmission opportunities. However, if done sequentially, this may take too much time to be practical (finding white space for communication is a real-time application), and parallel processing would be costlier, more complex and more energy-consuming. Note that sequential scanning allows to stop the process whenever an available band is identified, thus substantially speeding up the operation, except in the worst-case behavior where no band is available or where only one band is available and is scanned last.

CS enables to scan the wideband at once. Using CS principles, a dedicated sub-Nyquist sampling infrastructure can sample the signal in the wideband below the Nyquist rate, thus relaxing some constraints on the ADCs. Several such architectures are presented in Section 1.3.2.

The authors of [102] classify SS as a *support identification* problem. Indeed, in this application, we are not interested in recovering the signal itself, but rather its support *i.e.* the locations of active signal components in the frequency domain. In a naive reasoning, algorithms presented in Section 1.2.6 can be used to recover the wideband signal itself, and once the signal is recovered, estimating its support using spectrum analysis is fairly simple. However, this can be a

waste of time and computational resources, because several of these algorithms (particularly the greedy algorithms) first estimate the signal support, then use the estimated signal support to remove equations from the USLE and ultimately recover the signal itself. Skipping the last step (signal recovery from the estimated support) allows to save time and computational resources.

1.3.1.2 Detection of Active Devices in IoT networks

Many Internet of Things (IoT) devices, including sensors and monitoring devices, embrace a paradigm of communications different from H2H or H2M communications. More specifically, a great number of devices can be deployed in a limited geographical area, yet every single device transmits data quite rarely to the corresponding Access Point (AP), meaning that the number of active devices at any given time is much lower than the total number of devices.

Many IoT devices are subject to energy constraints and their messages are rather short[103]. As a consequence, conducting active user identification through message handshaking would be too cumbersome [104]. The problem of easily identifying active devices at the AP is of crucial importance in massive Machine Type Communications (mMTC), present in 5G wireless communications.

In mMTC, the high total number of devices makes it difficult to allocate orthogonal time-frequency resources to every single device. To mitigate this issue, non-orthogonal multiple access (NOMA) has been proposed [105]. Suppose the network hosts n devices and that a quasi-orthogonal signature (codeword) of length $m < n$ is assigned from a codebook $Q = \{q_1, \dots, q_n\}$). Since $m < n$, it is not possible to guarantee orthogonality between any two codewords (*i.e.* $\langle q_i, q_j \rangle \neq 0$ for $i \neq j$). Out of n devices in total, $k \ll n$ are trying to reach the AP at a given time. To do so, each device sends its signature codeword to the AP. In turn, the AP has to determine which devices are transmitting information to the AP. This step is called *active user detection* (AUD). Under the flat fading channel assumption, the received vector at the AP is of the form:

$$\begin{aligned} y &= \sum_{i=1}^{n} h_i q_i p_i + w \\ &= \begin{bmatrix} q_1 \dots q_n \end{bmatrix} \begin{bmatrix} h_1 p_1 \\ \vdots \\ h_n p_n \end{bmatrix} + w \\ &= Hs + w, \end{aligned} \tag{1.30}$$

where p_i is the symbol transmitted from the ith device, h_i is the scalar channel from the ith device to the AP, $H = [q_1 \dots q_n]$ is the $m \times n$ matrix generated using the codebook Q and w is the $m \times 1$ noise vector.

Because only $k \ll n$ devices are active, the vector s is sparse. Moreover, the set of active devices coincides with the set of indices of the non-zero elements of s (*i.e.* the support of s). As a consequence, CS-based AUD is another instance of a *support identification* problem and can be solved in a similar fashion. After the support of s is determined, an USLE can be derived from (1.30) by removing the rows of s and afferent columns of H that correspond to elements

not present in the support of s. This new USLE is overdetermined and can be solved using conventional Least-Squares.

1.3.1.3 Channel Estimation

Knowledge about the propagation channel of a transmission, or channel state information (CSI), is crucial to many signal processing algorithms. CSI acquisition, or channel estimation (CE), is often performed by sending known signals called pilots before along with communications. Because pilots constitute overhead (no user data is sent but time and power resources are consumed nonetheless), research has been carried out towards maximizing the efficiency of pilot-based CE.

Leveraging on the sparsity of the channel impulse response (CIR) is one way to increase the efficiency of pilot-based CE. In practice, in several multi-path scenarios, the number of propagation paths can be small. As a consequence, the communication channel has a sparse representation in the delay-Doppler domain [106] or in the angular domain [107]. CIR sparsity in the delay-Doppler domain is present in several channel models, like the ultra-wideband (UWB) channel [108], the underwater acoustic channel [102] or the extended typical urban (ETU) channel model in long term evolution (LTE). CIR sparsity in the angular domain is more prevalent in models for communications contexts with high spatial resolution properties, like millimeter wave [109] and MIMO [107].

Owing to CIR sparsity, CS techniques can be applied to CE to reduce the pilot-induced overhead. Let us consider the case where the CIR is modeled as $h = [h_1 \dots h_L]^T$ with $||h||_0 \ll L$. The channel is then convoluted with the known pilot sequence $a = [a_1 \dots a_P]^T$. Assuming $P > L$, the received signal is $y = [y_1 \dots y_{P-L+1}]^T$. A matrix system where the unknown sparse input is the CIR h can be constructed using the linear relation between y and h:

$$y = Ah + w, \tag{1.31}$$

where A is the Toeplitz matrix of size $(P - L + 1) \times L$ derived from the pilot sequence a:

$$A = \begin{bmatrix} a_L & a_{L-1} & \dots & a_1 \\ a_{L+1} & a_L & \dots & a_2 \\ \vdots & \vdots & \ddots & \vdots \\ a_P & a_{P-1} & \dots & a_{P-L+1} \end{bmatrix}. \tag{1.32}$$

If L and the length P of the pilot sequence verifies $P - L + 1 < L$, A corresponds to a CS matrix and (1.31) is an USLE solvable with CS techniques presented in Section 1.2.6. Note that random Toeplitz matrices satisfy the RIP under certain conditions [110], providing some CIR recovery guarantees.

CS-based methods for CE have outperformed conventional pilot-training CE techniques [111].

1.3.1.4 Other applications

These three applications (spectrum sensing, detection of active IoT devices and channel estimation) are only a handful of the possibilities enabled by CS. For additional material regarding the applications of CS to telecommunications, we refer the reader to the following surveys: [111, 102, 112].

1.3.2 Compressed Sensing Radiofrequency Architectures

Legacy, ubiquitous Nyquist sampling is, from a hardware perspective, well mastered because it relies on uniform sampling and common analog processing steps. As mentioned in the general introduction, Nyquist sampling is in practice limited by ADCs bounds on sampling rates [17]. Consequently, the CS theory provides key concepts to go beyond the limits of Nyquist sampling. However, not all the findings stemming from CS (some of which were presented in previous sections) can be directly implemented in telecommunications radio receiver architectures. Since the development of the CS theory, several CS-based architectures, also referred to as CS schemes in this document, have been proposed.

The principle common to all CS schemes is to sample a signal below the Nyquist rate and rely on the sparsity property of the signal to ultimately recover either the signal itself or some characteristic of it. Representations in which signal sparsity manifests itself are diverse and include the time, frequency, space, code and angle-of-arrival domains. Meanwhile, dictionaries Φ used to go from the original representation of the signal x (in which x is non-sparse) to a representation in which it is sparse include the DFT and DCT matrices as well as the wavelet transform. Here, we will assume signal sparsity in the frequency domain, and the considered sparsity dictionary is the DFT matrix. Furthermore, we will suppose that x is a multi-band signal, meaning that its energy in the frequency domain is contained in a finite union of closed intervals [113]. The author of [114] proposes to classify the various CS schemes into 3 categories:

- Non-Uniform Sampling (NUS);
- Variable Rate Sampling (VRS);
- Random Demodulation (RD).

In this subsection, we study one typical CS scheme per category. First, we present Multi-Coset Sampling, a NUS scheme. Then we turn our attention to Multi-Rate Sampling, a VRS scheme. Finally, we focus on the Modulated Wideband Converter, a RD scheme. For more exhaustive coverage of CS schemes, the reader is referred to the following surveys: [114, 115].

1.3.2.1 Multi-Coset Sampling

Multi-Coset Sampling (MCS) [116] or Periodic Non Uniform Sampling is a scheme in which several interleaved ADCs sample a wideband signal $x(t)$ under the Nyquist rate, as depicted in Fig. 1.5. Let $T = 1/f_{Nyq}$ denote the Nyquist period. The scheme consists in p branches, each of which contains an ADC that samples uniformly with a sampling period LT at a time different

from the ADCs of the other branches. The ADC of branch i samples at instants $t_k = kLT + c_iT$ where c_i is an integer that verifies $0 \leq c_i < L$. The sorted set $\mathcal{C} = \{c_i\}_{0 \leq i < p}$ is called a *pattern* of MCS. The sampling instants of MCS are depicted in Fig. 1.6.

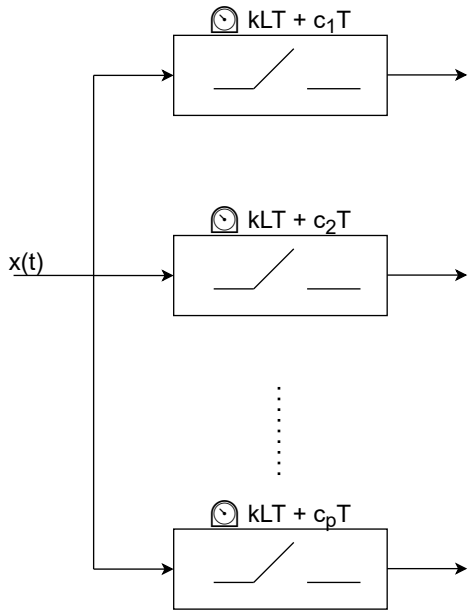


Figure 1.5: Description of MCS.

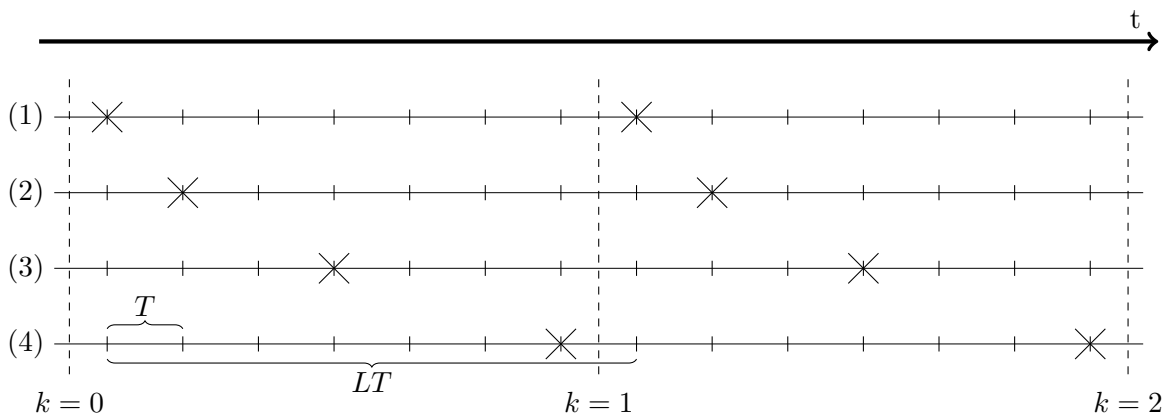


Figure 1.6: Multi-Coset sampling grid. The crosses represent the sampling instants $kLT + c_iT$ of MCS for $p = 4$ branches with $L = 7$ and $c_i = 0$, 1, 3 and 6 respectively. T is the Nyquist period.

MCS can also be described as Nyquist sampling where only $p < L$ samples out of every L Nyquist-rate samples are kept. The p kept samples are indexed by the values of $\mathcal{C}$. The sampling sequence of the ith sampling branch is described in [117] and reproduced here:

$$x_{c_i}[n] = \begin{cases} x(nT), n = mL + c_i, m \in \mathbb{Z} \\ 0 \text{ otherwise.} \end{cases} \tag{1.33}$$

Because each branch samples at a $1/LT$ rate, the signal, of total bandwidth $1/T$ Hz, is folded onto a $1/LT$ Hz wide bandwidth due to aliasing. If we represent the original frequency-

domain signal into L $1/LT$-Hz-wide slices, we can say that all spectrum slices are folded onto the central spectrum slice in each branch. Between the different branches, the variations are in the complex phases of the spectrum slices.

The Fourier transform $X_{c_i}(e^{j2\pi fT})$ of $x_{c_i}[n]$ is related to the unknown Fourier transform $X(f)$ of $x(t)$. This USLE can be expressed in a matrix form:

$$y(f) = Ax(f), \forall f \in [0, \frac{1}{LT}). \tag{1.34}$$

where $y(f)$ is a vector of length p whose ith element is $X_{c_i}(e^{j2\pi fT})$, A is the sampling matrix whose ikth coefficient is

$$A_{ik} = \frac{1}{LT} e^{j2\pi c_i k/L}, \tag{1.35}$$

and $x(f)$ is a vector of length L whose ith element is $X(f + \frac{i}{LT})$.

Parameter tuning The main parameters of the MCS are p, L and $\mathcal{C}$. If information about band locations is known [116], signal recovery is possible if $x(f)$ is p-sparse, $\forall f \in [0, \frac{1}{LT})$. The choice of these parameters determines the average sampling period of the MCS, which is $T_s = LT/p$ and its average sampling rate of $f_s = p/LT$.

Using the spark of A, it is shown in [116] that there are so-called *universal sampling patterns* for which $spark(A) = p$. Thanks to this property, signal recovery is guaranteed. Note that in the blind scenario, where band locations are unknown, $spark(A) = p$ would only guarantee recovery of $\frac{p}{2}$-sparse signals. Construction of universal sampling patterns is explored in [117, 118].

Another important practical aspect is the numerical stability of A. Indeed, sampling impairments, quantization errors, and noise can alter $y(f)$ in such a way that a poorly conditioned sampling matrix A could make signal recovery impossible in practice [119]. Consequently, the choice of $\mathcal{C}$ should also result in a well-conditioned sampling matrix A [120].

Advantages and drawbacks MCS is a simple and straightforward scheme that allows for a reduction of the average sampling rate and thus of the number of acquired samples. Furthermore, the post-processing step for signal recovery is rather simple [121], opening the way for a quick signal recovery, which is convenient in online applications like DSA.

A disadvantage of MCS is that a large analog bandwidth is still required for the ADCs, even if they do not sample at the Nyquist rate [122]. This is because there is no analog preprocessing step to reduce the received signal bandwidth prior to sampling. Furthermore, the shifters that maintain accurate delays between the different branches of the MCS require a Nyquist-rate clock for synchronizing the MCS branches [123]. Finally, the number of required MCS branches p to recover the active signal bands might be impractically high [115].

1.3.2.2 Multi-Rate Sampling

Before introducing Multi-Rate Sampling (MRS), we remind the reader of another sampling technique called undersampling.

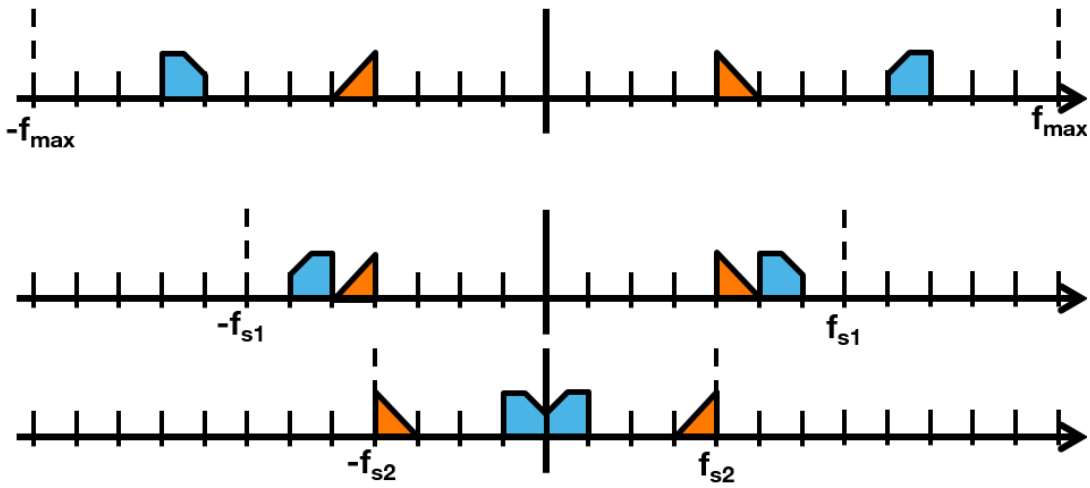


Figure 1.7: Undersampling a sparse multi-band signal at sampling rates $2f_{s1}$ and $2f_{s2}$ (signals represented in the frequency domain).

Undersampling Undersampling a signal consists in sampling it at a rate below the Nyquist rate. Doing so causes the spectral component(s) of the signal to fold and form alias(es). However, when acquiring the lower-frequency alias(es) of the signal component(s) instead of the original signal component(s), no information inside the signal component(s) is lost [122] except to possible alias interference and noise folding. By performing translation to baseband at no extra cost, undersampling turns aliasing from a phenomenon often seen as detrimental into an advantageous sampling method. The location in the frequency spectrum of the alias of a signal band depends on the sampling rate and on the location of the original band. Undersampling is generally poorly fitted for multi-band signals and for signals with unknown frequency support, as a badly chosen sampling rate will result in band aliases interfering with each other. Furthermore, while the information in an alias is the same as in the original band, the location of this band in the frequency spectrum cannot be determined based solely on the location of its alias. The adequacy of undersampling with regard to this property depends on the targeted application.

Fig. 1.7 depicts the undersampling of a real sparse signal made of two frequency components (and their symmetrical counterparts, due to the signal being real). On the first line, we see the frequency-domain representation of the signal sampled at the Nyquist rate $2f_{\max}$. On the second line, the signal is sampled at rate $2f_{s1} < 2f_{\max}$. As a consequence, the pentagon-shaped component (in light blue), whose original frequency is above f_{s1}, is folded onto the $[-f_{s1}, f_{s1}]$ interval, so what is sampled is actually an alias of this component. Meanwhile, the triangle-shaped component (in orange) is not folded. On the third line, the sampling rate $2f_{s2}$ is lower, and both the pentagon-shaped and the triangle-shaped components are folded. Note that the position of the alias of the pentagon-shaped component differs depending on the sampling rate.

As shown in Fig. 1.7, if several samplers on different branches undersample the same signal at different rates, the location of the aliases in the frequency spectrum will differ from branch to branch, opening the way for support recovery and full signal recovery: this is the principle of the MRS.

Presentation of the MRS scheme MRS is a CS scheme in which the received signal goes through L branches, each of which contains an ADC sampling uniformly at a rate below Nyquist, as depicted in Fig. 1.8. Essentially, undersampling is carried out in each branch. The sampling rates of the different ADCs are different from one another. As a consequence, the frequency-domain representation of the sampled signal is different from one branch to another. More precisely, the various representations display the same artefacts (possibly mirrored) but in different locations of the spectrum.

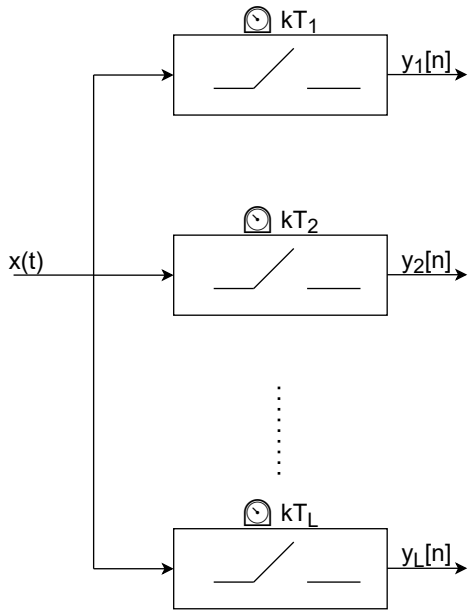


Figure 1.8: Description of MRS.

MRS comes in two flavors: Asynchronous MRS (AMRS) [113] and Synchronous MRS (SMRS) [20]. In AMRS (resp. SMRS), the time difference between the sampling instants of the different branches is unknown (resp. known).

In AMRS, the envelope of the signal, as well as its frequency support, can be recovered from the observations in the different branches. However, the observations cannot be combined to recover the signal phase. Note that the signal phase can still be retrieved if for each signal band, there is at least one branch in which the signal band does not interfere with other aliases. The major advantage of AMRS is that no tight synchronicity between the different branches' ADCs is required. This seems particularly adapted to the multi-site architecture presented in Section 0.2.3 of the general introduction, due to the possible relaxed synchronization between geographically-distant sites.

In SMRS, both the signal envelope and phase can be recovered from the combined observations (provided that the corresponding USLE can be solved somehow). However, a tight synchronization between the different branches' ADCs is needed, which can in practice require a clock running at a much higher speed than each ADC's clock.

In the remainder of this manuscript, we will focus on SMRS and refer to it as MRS. Nonetheless, since AMRS and SMRS are fairly similar, many findings concerning SMRS are also applicable to AMRS.

Mathematical description of the MRS Let $x(t)$ be a sparse real-valued continuous baseband signal, whose support (set of frequencies where x is non-zero) is comprised within $[-f_{max}, f_{max}]$. The $L \geq 1$ branches of a Multi-Rate Sampler (MRS) each sample $x(t)$ uniformly at a sampling rate $\nu_i, 0 \leq i < L$ (sampling period $T_i = 1/\nu_i$). The resulting sampled signals are given by

$$y_i(t) = \sum_{j=-\infty}^{+\infty} x(t)\mathbb{1}_0(t - \frac{j}{\nu_i}),\ 0 \leq i < L, \tag{1.36}$$

where $\mathbb{1}_0(z)$ is the indicator function of the $\{z = 0\}$ set.

Let Δ be the duration of the time window during which the signal is sampled and $M_i = \Delta\nu_i$ (resp. $N = \Delta\nu_{Nyq}$) be the number of samples collected by branch i (resp. collected by an hypothetical branch sampling at the Nyquist rate) during the time window (see Fig. 1.9). Since M_i and N are integers, the time window duration Δ and the sampling rates ν_i must be chosen with care. From now on, y_i (resp. x) will refer to a discrete sample vector of length M_i (resp. N). N is called the block size. Let $\delta_i = M_i/N \leq 1$ be the *undersampling ratio* at branch i. Fig. 1.9 depicts an example of sampling grid for MRS.

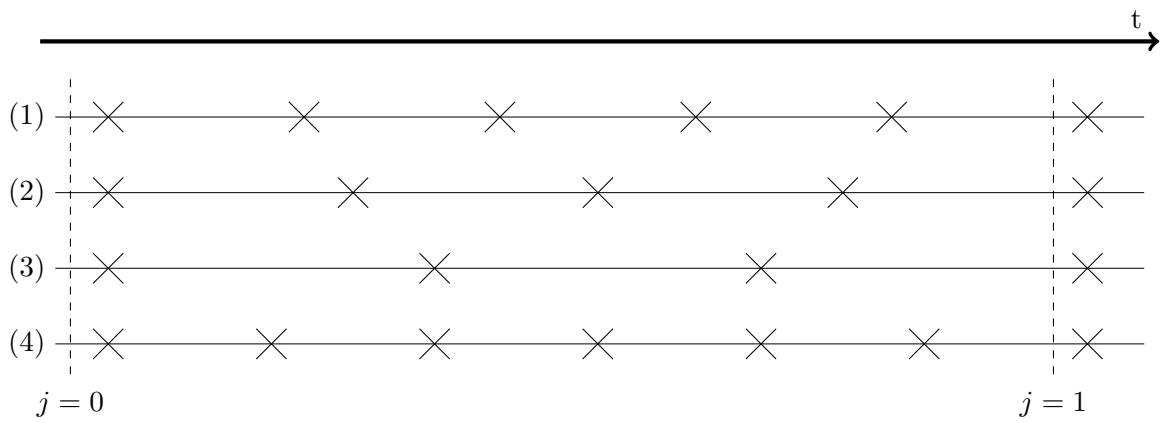


Figure 1.9: Multi-Rate sampling grid. This grid displays the Dirac impulses (sampling instants) for $L = 4$ branches with $M_i = 5$, 4, 3 and 6 respectively.

Undersampling can be represented as $Y_i = F_i X$ in the frequency domain, where F_i is a $M_i \times N$ *folding matrix* and Y_i (resp. X) is the Fourier Transform of y_i (resp. x).

The coefficients of F_i have been described in [20] and can be rewritten as follows. If N is odd (resp. even), the coefficients of F_i are obtained as such:

$$f_{jl} = \begin{cases} \delta_i \mathbb{1}_0((j-l) \bmod M_i), & \text{if } l \leq \frac{N-1}{2} \text{ (resp. } l < \frac{N}{2}) \\ \delta_i \mathbb{1}_0((j-l+N) \bmod M_i), & \text{if } l > \frac{N-1}{2} \text{ (resp. } l > \frac{N}{2}) \end{cases}. \tag{1.37}$$

Additionally, if N is even and $l = \frac{N}{2}$, we have

$$f_{jl} = \frac{1}{2}\delta_i(\mathbb{1}_0((j-l) \bmod M_i) + \mathbb{1}_0((j-l+N) \bmod M_i)). \tag{1.38}$$

Next, the observations Y_i and folding matrices F_i are concatenated to yield the following

system:

$$Y = AX, \text{ where } Y = \begin{bmatrix} Y_0 \\ \vdots \\ Y_{N-1} \end{bmatrix} \text{ and } A = \begin{bmatrix} F_0 \\ \vdots \\ F_{N-1} \end{bmatrix}. \tag{1.39}$$

Here, Y is the measurement vector of size $M = \sum_i M_i < N$, A is the sampling matrix of size $M \times N$ and X is still of size N.

The USLE (1.39) can be solved directly using the hypothesis that X is sparse. However, to reduce the complexity of USLE resolution, a reduction procedure is proposed in [20]. The main idea behind the reduction procedure is the following: a down-converted signal band's location in the frequency spectrum determines a limited set of possible locations for the original band's location. Furthermore, when a signal is undersampled at different rates, the locations of the alias of a signal band differ, and so do corresponding up-converted aliases. The only up-converted alias identically located no matter the sampling rate is the one corresponding to the location of the original band. This simple and straightforward reduction procedure can be done without numerical system resolution. It is the main method for signal support recovery in AMRS [113] and is also a crucial step to SMRS. Indeed, identifying the signal support (or a set of intervals that contain the support, if some uncertainty remains after the reduction procedure) allows to set variables of the USLE (1.39) not located in the signal support to zero. Doing so reduces the number of variables, but the number of equations remains unchanged. Consequently, the reduced USLE is easier to solve, and depending on the level of sparsity of x and the number of measurements M, the reduced problem can even be overdetermined. An overdetermined problem is much easier to solve than an underdetermined problem.

Parameters of MRS The main parameters of MRS are the number of branches L and the marginal sampling rates M_i. In [124], it is suggested that the $\{M_i\}_i$ be different primes. We will improve this contribution in Chapter 3. Note that the authors of [113, 20] advise to have a limited number L of branches (up to 5-10 branches) while the authors of [124], who focus on AMRS applied to signal energy detection, routinely use hundreds of branches. A limited number of branches seems more adequate for practical implementation. Furthermore, having a lot of branches and a low overall number of measurements M requires each branch to sample at a rate M_i several orders of magnitude below Nyquist, which can cause a lot of interference for aliased signals (especially is the occupancy ratio k/N of a signal is high).

Advantages and drawbacks Because ADCs sample uniformly at a rate lower than Nyquist, the MRS, and especially its asynchronous flavor, has a low implementation complexity. It also features fewer branches than MCS. MRS entails a fairly straightforward post-processing step. It is robust with not very sparse input signals [114]. However, as in MCS, a large analog bandwidth is required for MRS, and synchronization (in the synchronous flavor) can be challenging [115].

1.3.2.3 Modulated Wideband Converter

The Modulated Wideband Converter (MWC) [125] is a CS scheme inspired from previous works on Random Demodulation [126]. MWC features m parallel branches. In each of these branches, the input signal is analogically mixed with a periodic sequence of pseudo-random ± 1's varying at a rate higher or equal to the Nyquist rate f_{Nyq}, then goes through an ideal low-pass filter, and is finally sampled uniformly at a low rate f_s. The periodic mixing step allows for the RF signal bands to be folded at baseband, which in turn permits low-rate sampling. Note that the ideal low-pass filter can be substituted by any filter satisfying the Nyquist intersymbol interference (ISI) criterion [127]. Fig. 1.10 depicts the structure of the MWC.

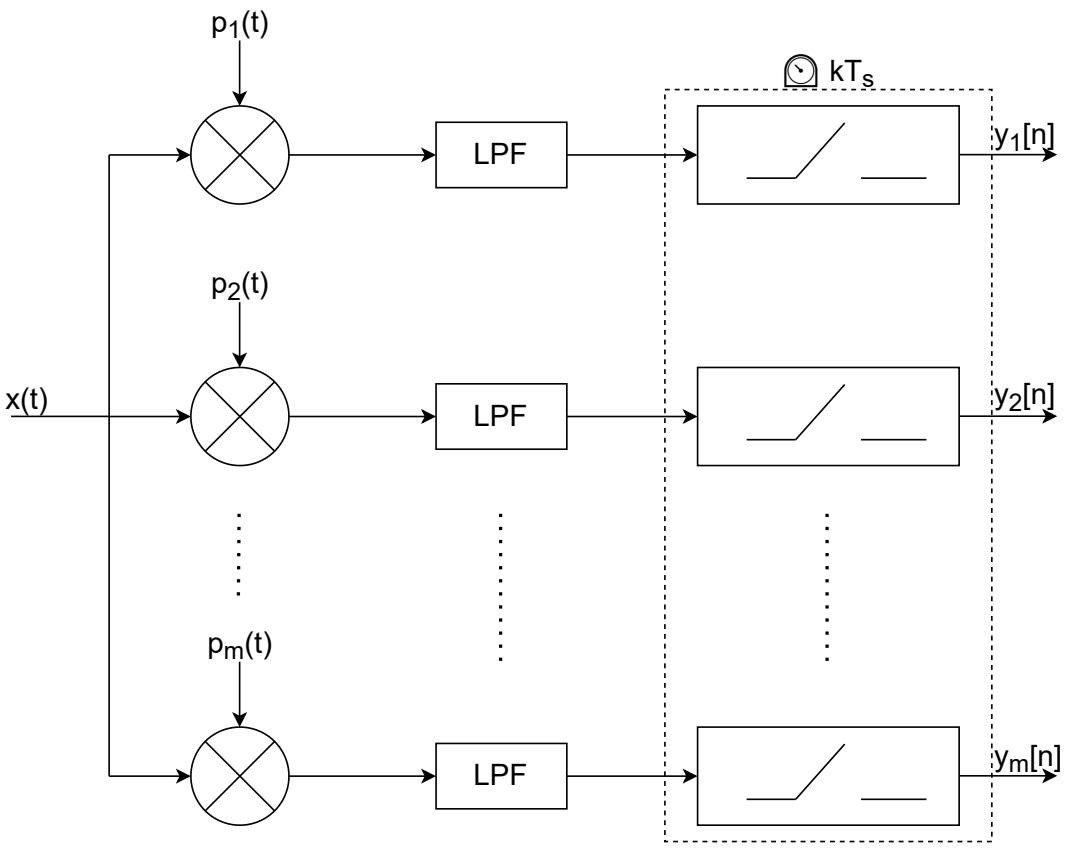


Figure 1.10: Description of the MWC.

In the MWC, the diversity is brought by the different mixing functions $p_i(t), 0 \leq i < m$ with which the signal is multiplied. Indeed, these functions determine which signal bands are active in the aliased sampled signal. Resulting linear equations can be combined to form an USLE, which can be solved using signal recovery techniques presented in Section 1.2.6.

Mathematical Description of the MWC In each branch, the input signal $x(t)$ is multiplied by a mixing function $p_i(t), 0 \leq i < m$. The mixing functions are T_p-periodic and contain M ± 1's in each period. An example of a mixing function is depicted in Fig. 1.11. After being multiplied by a mixing function, the signal goes through a low-pass filter with cut-off frequency $1/(2T_s)$, where T_s is the sampling period. The filtered signal is subsequently sampled by conventional

ADCs with sampling period $T_s > T_{Nyq}$ to yield $y_i[n]$, the sequence of samples acquired at branch i.

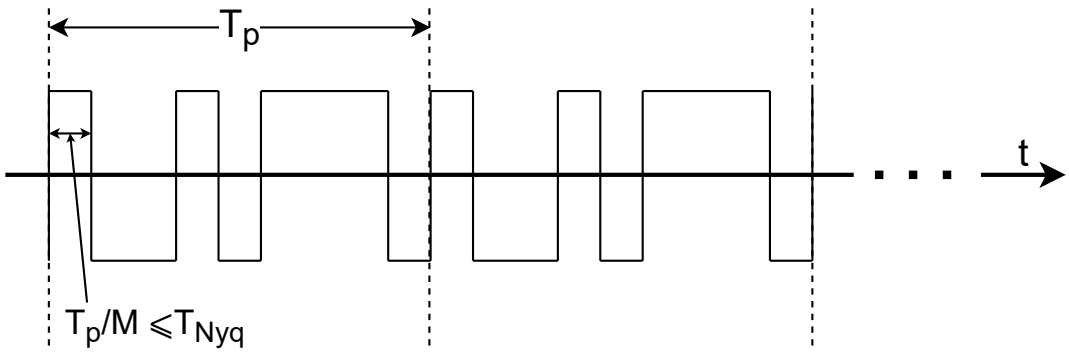


Figure 1.11: Mixing function $p_i(t)$.

In order to explicit the relationship between sample sequences $y_i[n]$ and the unknown signal $x(t)$, we reproduce the reasoning presented in [125]. Let us introduce the following:

$$f_p = 1/T_p,\ \mathcal{F}_p = [-f_p/2, f_p/2], \tag{1.40}$$

$$f_s = 1/T_s,\ \mathcal{F}_s = [-f_s/2, f_s/2]. \tag{1.41}$$

We focus on the ith channel. Since $p_i(t)$ is periodic, it accepts the following Fourier decomposition:

$$p_i(t) = \sum_{l=-\infty}^{\infty} c_{il} e^{j2\pi lt/T_p}, \tag{1.42}$$

where the Fourier coefficients c_{il} follow:

$$c_{il} = \frac{1}{T_p} \int_0^{T_p} p_i(t) e^{-j2\pi lt/T_p} \mathrm{d}t. \tag{1.43}$$

The Fourier transform of the analog multiplication $\tilde{x}_i(t) = x(t)p_i(t)$ is

$$\begin{aligned} \tilde{X}_i(f) &= \int_{-\infty}^{\infty} x(t) \left(\sum_{l=-\infty}^{\infty} c_{il} e^{j2\pi lt/T_p} \right) e^{-j2\pi ft} \mathrm{d}T \\ &= \sum_{l=-\infty}^{\infty} c_{il} X(f - lf_p) \end{aligned} \tag{1.44}$$

Since $X(f) = 0$ for $f \notin [-f_{Nyq}/2, +f_{Nyq}/2]$, the sum in (1.44) contains a finite number of nonzero terms.

The signal $\tilde{x}_i(t)$ now passes through an ideal low-pass filter with cut-off frequency $f_s/2$. As a consequence, the filtered signal only contains the frequencies present in the interval $\mathcal{F}_s$. The discrete-time Fourier transform (DTFT) for sample sequence $y_i[n]$ can be written as:

$$\begin{aligned} Y_i(e^{j2\pi fT_s}) &= \sum_{n=-\infty}^{\infty} y_i[n] e^{-j2\pi fnT_s} \\ &= \sum_{l=-L_0}^{+L_0} c_{il} X(f - lf_p),\ f \in \mathcal{F}_s, \end{aligned} \tag{1.45}$$

where L_0 is chosen to be the smallest integer so that the sum in (1.44) contains every nonzero contributions of $X(f)$ over $\mathcal{F}_s$.

Eq. (1.45) links known sequences $y_i[n]$ (or more exactly, their DTFTs) to the unknown input signal $X(f)$. This equation can be written in matrix form so as to match the typical USLE (1.15) of the CS framework:

$$y(f) = Az(f),\ f \in \mathcal{F}_s, \tag{1.46}$$

where $y(f)$ is a vector of length m whose ith element is $y_i(f) = Y_i(e^{j2\pi f T_s})$.

The unknown vector $z(f) = [z_1(f), \ldots, z_L(f)]^T$ is of length $L = 2L_0 + 1$ (a closed-form expression to evaluate L_0 is proposed in [125]), and its ith element is given by

$$z_i(f) = X(f + (i - L_0 - 1)f_p),\ 1 \leq i \leq L,\ f \in \mathcal{F}_s. \tag{1.47}$$

Finally, the $m \times L$ matrix A contains the coefficients c_{il}:

$$A_{il} = c_{i,-l} = c_{il}^* \tag{1.48}$$

Advantages and drawbacks The main difference between MWC and the previously mentioned schemes (MCS/MRS) is the presence of the analog mixing step. Because all ADCs sample uniformly and simultaneously, synchronization is easy (only one low-rate clock is required). Furthermore, since the analog mixing folds all frequencies to baseband, the inputs of the ADCs have a narrow (analog) bandwidth, so the need for a large ADC analog bandwidth is alleviated.

The analog mixing codes run at least at the Nyquist rate. While there is no particular hardware implementation limitation to this [128], this step is fairly energy-intensive [114]. Furthermore, because the MWC assumes a static spectrum over a long period of time, time-dependant input like Radar pulses or short IoT messages can alter the performance of MWC [129]. Finally, a bottleneck for the MWC resides in the mixing codes, which need to be selected carefully. A review of popular mixing codes is provided in [130].

1.3.3 Conclusion on the Applications and Architectures of CS

In this section, we have described a few applications of CS in telecommunications, as examples of how this recent yet rich framework can be used to make telecommunications more efficient. We have also described the main radiofrequency architectures used for CS-based signal sampling. It is important to keep in mind that signal sparsity in telecommunications can manifest itself in a variety of ways. This underlines the potential of CS when it comes to sampling telecommunications signals, compared to the Nyquist approach to sampling.

What was still a theoretical subject of interest just over a decade ago is evermore present in today's communications. CS is in phase with many current concerns, such as the struggle against spectrum scarcity, or the energy consumption issue with wideband sampling. As a consequence, its further expansion in future communications systems would not come off as a surprise.

We now turn to the study of spectrum sensing, a subject that somewhat intersects CS (see Section 1.3.1.1) and offers promising solutions for some of the industry's aforementioned issues.

1.4 Spectrum Sensing

As mentioned in Section 1.3.1.1, spectrum sensing (SS) is a core component of the CR. In this section, we will first introduce SS and the closely related problem of Dynamic Spectrum Access (DSA). We will also see how SS can be used for our purpose of efficiently sampling IoT communications that are sparse in time and/or frequency.

SS can be decomposed in two subproblems: narrowband SS (where only one frequency channel is sensed at a time) and wideband SS (several frequency channels are sensed at the same time). We will first focus on the signal processing techniques that have been applied to narrowband SS. Following the study of narrowband SS, we will turn to wideband SS, present the main techniques used for this subproblem, and see how it can be used jointly with CS.

1.4.1 Introduction: Dynamic Spectrum Access and Sampling Infrastructure

1.4.1.1 Dynamic Spectrum Access

The radio spectrum has historically been managed in a static fashion, with regulating bodies like the FCC in the USA or the ANFR in France allocating operating licenses to users based on their specified requirements. However, a large part of the electromagnetic spectrum has now been allocated, offering little to no leverage for the introduction of new users. Still, the number of use cases, users and devices relying on wireless communications is growing, exacerbating the problem known as *spectrum scarcity*. A way to combat spectrum scarcity is to reallocate unused spectrum bands. Consider the "digital dividend" bands [131]: at the turn of the 2010s, the analog TV bands were switched off and the freed spectrum was reallocated to various users (digital TV, 4G communications, and so on). However, this solution is often tedious to implement. Another solution to fight spectrum scarcity is to find new spectrum bands on which to operate. An example planned in 5G is communications using millimeter waves (mmWave). The development of communications on new bands is often hindered by unfavorable physical channel characteristics, such as absorption by atmospheric gases for mmWave [132]. It is also subject to hardware limitations, such as ADC sampling rate requirements or synchronization issues. Consequently, extending the span of operable frequencies is more to be seen as a long-term improvement than an immediate cure-all for spectrum scarcity.

A third approach to reduce the detrimental effects of spectrum scarcity is to improve spectral efficiency. While using spectral resources more efficiently is often carried out through more elaborate modulations, a simpler consideration can incur dramatic gains. Measurement campaigns over the last decades have shown that the licensed spectrum is often underutilized, meaning that some licensed frequency bands are temporarily vacant [133, 134]. This has led to the rise of dynamic spectrum management. Regulatory frameworks like Licensed Shared Access (LSA) [135] in Europe or Citizens Broadband Radio Service (CBRS) [136] in the USA have been pro-

posed in order to allow voluntary spectrum sharing. While the specifics of each shared spectrum scheme slightly differ from one another, all of them are governed by three principles:

1. There are at least two categories of users, the "primary users" (PU) having primacy over "secondary users" (SU) for access to the resource;

2. SUs can use the resource as long as they cause no interference to PUs;

3. A SU is required to back off from the resource as soon as i) PU activity is detected by the SU or ii) a PU request for medium access is received by the SU.

As we can see, these schemes are hierarchy-based and thus differ from horizontal MAC schemes like listen-before-talk or ALOHA.

To be able to transmit data, the SU needs to know the status of the resource (either vacant or occupied), and this knowledge is typically acquired through SS. (Other means like transmission schedules can be relevant in some use cases but are usually not very scalable, as the size of the table where the schedules are saved grows in a polynomial fashion.) As a consequence, SS needs to be fairly accurate. If SS detects a frequency channel to be free when it is not, subsequent SU transmissions will cause interference to the communications of the PU, resulting in possible sanctions to the SU. If SS detects that a channel is occupied when it is not, the SU will miss out on a transmission opportunity.

1.4.1.2 Infrastructure for Efficient IoT Sampling

Our proposed infrastructure (see section 0.2.3 of the General Introduction) can also greatly benefit from SS. We aim to sample in a blind fashion, meaning that we do not know the locations of the transmitted messages in the frequency spectrum before sampling. As a consequence, even if we have an idea of the level of spectral occupancy of the signal, we have to take more samples than if we were in a non-blind scenario, where the locations of the transmitted messages in the frequency spectrum are known before sampling. For example, for the MCS scheme presented in Section 1.3.2.1, recovering a blindly-acquired signal requires twice as many samples as recovering a non-blindly acquired signal [117].

Once samples are collected using a CS scheme, we wish to reduce the quantity of samples to store in the database as much as possible. One way to proceed is to keep the samples corresponding to the transmitted signal of interest and discard the other samples, as shown in Fig. 1.12. SS is appropriate for this purpose, as it takes a decision on presence or absence of signal of interest from a sequence of samples.

Many SS techniques have been developed with DSA in mind, yet our proposed application of SS is different from DSA. Two characteristics seem particularly relevant to us, for any given SS technique to be used for our proposed application:

- Our application does not require live processing, contrary to DSA. In this regard, our application has a relaxed requirement over DSA – although live processing enables to make our sampling system adaptive to changes in spectral occupancy over time for example.

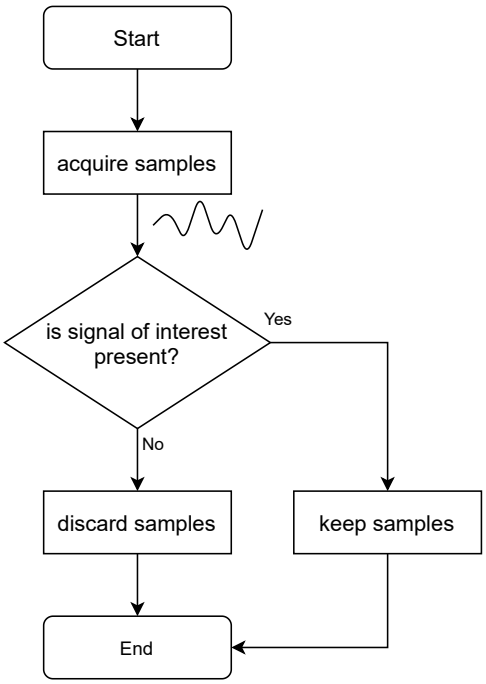


Figure 1.12: High-level flowchart describing the acquisition-storage part of our proposed infrastructure. The test for presence of signal of interest is carried out using SS techniques.

- DSA aims at finding *any* white space in the resource for transmission, while our proposed application seeks to find *all* occupied frequency channels. For wideband SS, as we will see in Section 1.4.3, scanning the entire band (required for our proposed application) corresponds to the worst-case scenario for some techniques, especially the sequential ones. As a consequence, these techniques are not to be favored for our proposed application.

1.4.1.3 Narrowband versus Wideband SS

According to the authors of [1], there are two categories in which a SS method can be classified: narrowband (NB) and wideband (WB) (see Fig. 1.13 for a classification of SS techniques). In NB-SS, test for presence of PU transmissions is carried out in one frequency channel at a time. In Section 1.4.2, we will present NB-SS techniques belonging to one of four families: energy detection, cyclostationary detection, matched filter detection, and covariance-based detection. Additionally, machine learning-based sensing can be used for NB-SS, but it will not be covered in the remainder of this manuscript: a review of machine learning-based approaches to SS is provided in [1].

An extension to NB-SS, WB-SS focuses on analyzing several frequency channels at a time. According to the authors of [137], WB-SS techniques can be categorized based on whether they rely on Nyquist-rate sampling or sub-Nyquist sampling. Nyquist-sampling WB-SS approaches include performing NB-SS on each frequency channel, whether sequentially or simultaneously while sub-Nyquist WB-SS techniques rely on the use of the CS framework. The principal approaches to WB-SS are presented in 1.4.3.

NB-SS and WB-SS are quite different, as can be inferred from Fig. 1.14. In NB-SS, we wish

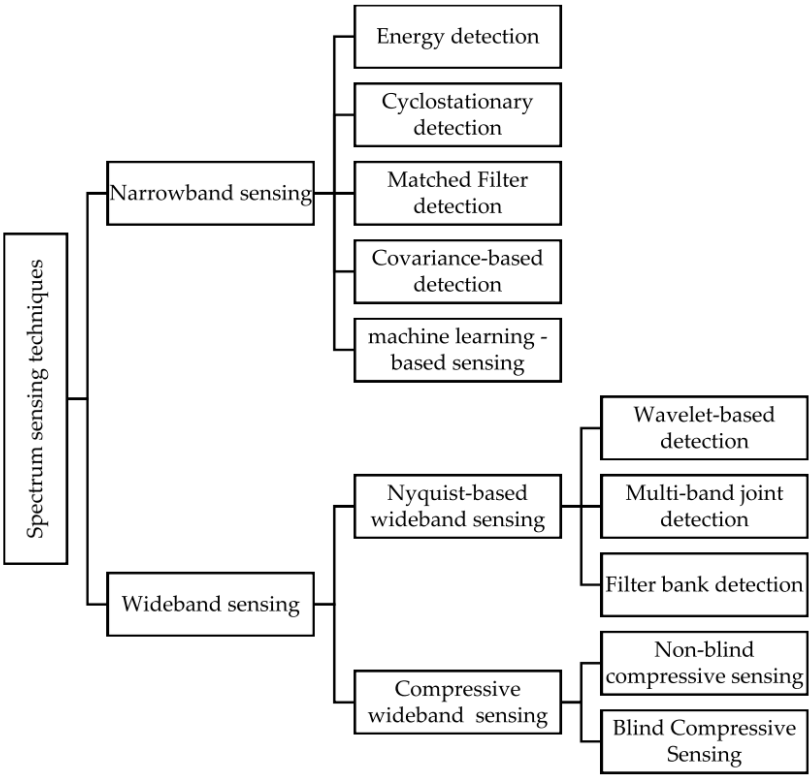


Figure 1.13: Classification of spectrum sensing techniques, as proposed by the authors of [1].

to know whether PU signal is present in the narrowband signal. In WB-SS, we also seek to locate the positions of PU signal in the frequency spectrum of the wideband signal.

1.4.2 Narrowband Spectrum Sensing

The aim of NB-SS is to decide whether a signal of interest is present in a sequence of samples of the received signal. In this regard, it is an application of binary hypothesis testing. Under the null and alternative hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$, the received signal is respectively modeled as follows:

$$\mathcal{H}_0 : y[n] = w[n], \tag{1.49}$$

and:

$$\mathcal{H}_1 : y[n] = s[n] + w[n], \tag{1.50}$$

where $y[n]$ is the received signal, $s[n]$ is the transmitted signal and $w[n]$ is additive white Gaussian noise with zero mean and a variance of σ_w^2.

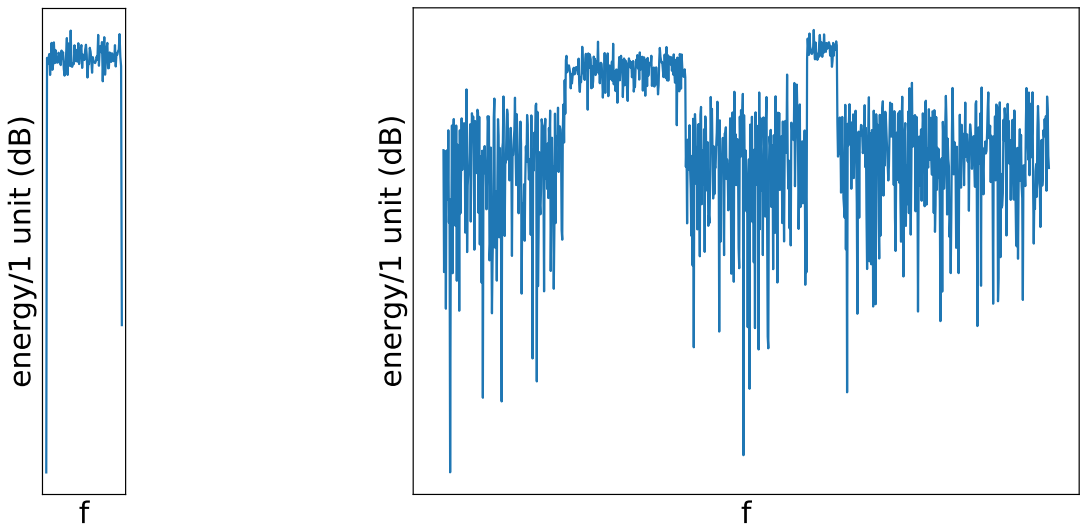
The null (resp. alternative) hypothesis corresponds to an absence (resp. presence) of signal of interest in the received signal.

Two widely used metrics to evaluate the performance of NB-SS techniques are the probabilities of correct detection P_D and the probability of false alarms P_{FA}, defined as follows:

$$P_D = P(\mathcal{H}_1|\mathcal{H}_1), \tag{1.51}$$

and:

$$P_{FA} = P(\mathcal{H}_1|\mathcal{H}_0), \tag{1.52}$$



(a) Narrowband signal.

(b) Wideband signal.

Figure 1.14: Power spectrum densities (PSD) of a (a) NB; (b) WB signal (semilog scale).

where $P(\mathcal{H}_i|\mathcal{H}_j), (i,j) \in \{0,1\}^2$ denotes the probability that $\mathcal{H}_j$ is true and $\mathcal{H}_i$ is decided.

1.4.2.1 Energy Detection

In energy detection (ED), a test statistic $T_{ED}(Y)$ is compared to a threshold η which depends on the noise characteristics:

$$T_{ED}(Y) \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \eta. \tag{1.53}$$

Under the Neyman-Pearson (NP) theory, the test statistic $T(Y)$ is proportional to the energy of Y [138]:

$$T_{ED}(Y) = \frac{1}{N}\sum_{n=1}^{N} |Y[n]|^2, \tag{1.54}$$

where N is the total number of samples and Y denotes the discrete Fourier transform of the received signal y.

The choice of the threshold η has a major impact on detection performance. If η is low (resp. high), the probability of correct detection P_D will be high (resp. low), and so will the probability of false alarms P_{FA}.

Suppose the transmitted signal $s(n)$ in (1.50) is a Gaussian random process with zero mean and variance σ_s^2. From (1.54) and the definition of $s(n)$ and $w(n)$, the test statistic $T_{ED}(Y)$ is a sum of N squares of independant Gaussian random variables. Consequently, it follows a central Chi-square distribution under hypothesis $\mathcal{H}_0$ and a non-central Chi-square distribution under hypothesis $\mathcal{H}_1$ [139]. For a large N, the central limit theorem applied to $T_{ED}(Y)$ holds and we have [140]:

$$\begin{cases} \mathcal{H}_0 : T_{ED}(Y) \sim \mathcal{N}(N\sigma_w^2, 2N\sigma_w^4) \\ \mathcal{H}_1 : T_{ED}(Y) \sim \mathcal{N}(N(\sigma_w^2+\sigma_s^2), 2N(\sigma_w^2+\sigma_s^2)^2) \end{cases}, \tag{1.55}$$

From (1.51), (1.52), (1.53) and (1.55), we obtain [140]:

$$P_D = \mathcal{Q}\left(\frac{\eta - N\sigma_w^2(1+\gamma)}{\sqrt{2N}\sigma_w^2(1+\gamma)}\right), \tag{1.56}$$

and

$$P_{FA} = \mathcal{Q}\left(\frac{\eta - N\sigma_w^2}{\sqrt{2N}\sigma_w^2}\right), \tag{1.57}$$

where $\mathcal{Q}(\cdot)$ is the complementary distribution function of the standard Gaussian and γ is the SNR defined as σ_s^2/σ_w^2.

From (1.56) and (1.57), P_D can be rewritten as:

$$P_D = \mathcal{Q}\left(\frac{1}{1+\gamma}\left[\mathcal{Q}^{-1}(P_{FA}) - \gamma\sqrt{\frac{N}{2}}\right]\right). \tag{1.58}$$

In the NP theory, η, P_D, P_{FA} and N are thus linked through the closed-form expression (1.58). For example, η can be set to a value associated in turn to a given, fixed false alarm rate. However, this closed-form expression also depends on the AWGN variance σ_w^2, which is generally unknown beforehand.

A straightforward way to estimate the AWGN variance is to estimate the channel statistics when the channel is free of transmissions [141]. The immediate caveat of this approach is the absence of certitude that the channel is free of transmissions without prior knowledge of the channel state information. The author of [2] estimates the AWGN variance by separating noise realizations which follow a Gaussian distribution from signal-of-interest realizations which are considered as outliers from the aforementioned Gaussian distribution. In [142], the AWGN variance is estimated by computing autocorrelations on the noisy received signal. The authors of [143] propose a double-threshold system to deal with the intermediate space where it is unsure whether there is a PU signal or not.

Advantages and Drawbacks of ED ED is extremely popular for two main reasons. First, it is computationally simple. Second, it requires no prior knowledge about the transmitted signal. A downside of ED's simplicity is that it cannot perform identification of signal components based on some intrinsic characteristic. Furthermore, its performance is heavily degraded under low SNRs. Finally, it requires a correct estimation of the noise variance. In Chapter 2, we will present a new method for AWGN variance estimation, which is suited to the class of signals we are interested in, that is, sparse multi-band signals.

1.4.2.2 Cyclostationary Detection

A signal generated by a process whose statistical characteristics vary periodically with time is said to be *cyclostationary*. Virtually all telecom signals are cyclostationary, thanks to periodic statistics such as modulation rate or carrier frequency. Meanwhile, most noise is stationary (*i.e.* its statistical characteristics do not vary with time). Detecting cyclostationary processes makes it possible to discriminate communications signals from noise and is the key enabler to cyclostationary detection (CD).

For CD, a received discretized signal $y[n]$ is considered cyclostationary if its mean and autocorrelation are periodic. Consider the instantaneous autocorrelation function:

$$r_y(n, \nu) = E\{y[n]y^*[n+\nu]\}. \tag{1.59}$$

If $y[n]$ is cyclostationary, $r_y(n, \nu)$ is periodic in n and can be decomposed using a Fourier transform. The coefficients of this decomposition are given by the cyclic autocorrelation function:

Definition 1.11 (Cyclic Autocorrelation Function)**.**

$$\tilde{r}_y(\alpha, \nu) = \frac{1}{N} \sum_{n=0}^{N-1-\nu} r_y(n, \nu) e^{-j2\pi\alpha n \Delta_s}. \tag{1.60}$$

Δ_s is the sampling period and α is called a cyclic frequency or cycle.

By performing a Fourier transform on the cyclic auto-correlation function over the time lags ν, the spectral correlation function is obtained:

Definition 1.12 (Spectral correlation function or cyclic spectrum)**.**

$$s_y(\alpha, f) = \sum_{\nu=0}^{N-1} \tilde{r}_y(\alpha, \nu) e^{-j2\pi f \nu \Delta}. \tag{1.61}$$

The cyclic spectrum is a function of two variables, f and α, whereas the power spectrum is a function of only f: this makes the cyclic spectrum richer than the power spectrum, but also more expensive to compute.

After the cyclic autocorrelation function is estimated [144], detection is carried out. A widespread test for detection is that of Dandawaté & Giannakis [145]: for a given cycle $\alpha \neq 0$, a signal is said to be present if there is at least one time lag ν so that $\tilde{r}_y(\alpha, \nu)$ is greater than some error threshold.

Advantages Cyclostationarity-based detection methods are good at discriminating cyclostationary processes from stationary processes such as noise. As such, in the low SNR regime, cyclostationarity-based methods outperform energy-based methods [115], which model signals as wide-sense stationary. Furthermore, they are also robust to noise level uncertainty [146].

Drawbacks A limitation is that computing the cyclic spectrum is more computationally demanding than the usual frequency spectrum, especially without knowledge of the cyclic frequencies to test.

Another drawback is the acquisition time, which can be long in the low SNR regime (the lower the SNR, the more samples needed to discriminate signal from ambient noise). This can be impractical for live scenarii in which a decision on the presence of signal needs to be done (quasi) instantaneously, such as SS for DSA. However, this constraint is relaxed in our scenario, since the signal information is to be retrieved and processed later in time.

Finally, the phenomenon of cyclic frequency mismatch, due to unknown errors at the transmitter clock, is detrimental to CD; however, it can be mitigated by using a Slepian basis expansion instead of the Fourier basis expansion previously mentioned [147].

1.4.2.3 Pilot-based Matched-filter Detection

Pilot-based matched-filter detection (MFD) [148, 149] is based on pilot sequences sent by the transmitter and known by the receiver beforehand. Using the pilot sequences, the following test statistic is computed on the sequence of samples $y[n]$:

$$T_{MFD} = \frac{1}{N}\sum_{n=1}^{N} y[n]x_p^*[n], \tag{1.62}$$

where N is the number of samples and x_p are the pilot samples. This test statistic is compared to a threshold η:

$$T_{MFD}(Y) \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \eta. \tag{1.63}$$

The particularity of MFD is its reliance on pilot sequences: the detector *must* know the pilot sequences used by the transmitter. In the context of SS, it can be a stark constraint, although some specific use cases are suitable for MFD, *e.g.* DSA on licensed bands used by mono-application primary users, like TV bands. If the pilot sequence knowledge constraint is met, MFD is efficient: highly-performing detection can be carried out with a small number of samples.

As in ED, the choice of the threshold has a significant influence over the outcome of the detection. The authors of [149] have proposed a method to select a dynamic threshold for MFD.

1.4.2.4 Covariance-based Detection

In covariance-based detection (CovD) [150, 151], the structure of the sample covariance matrix of the received signal is evaluated for presence of a primary user. This is possible because the sample covariance matrices of signal and noise are generally different (this difference being a necessary condition for CovD).

While all CovD techniques exploit the structure of covariance matrices, they use different approaches and test statistics. The authors of [150] proposed to compute the ratio between two test statistics defined as follows:

$$T_1 = \frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{N} |r_{nm}|, \tag{1.64}$$

and

$$T_2 = \frac{1}{N}\sum_{n=1}^{N} |r_{nn}|, \tag{1.65}$$

where r_{ij} is the ith-row, jth-column element of the sample covariance matrix of the received signal R_y and N is the number of consecutive samples. Here a sample is defined as a N-sized vector whose elements consecutive values of the received signal, meaning that in the usual sense of the word "sample" in this manuscript, this method takes N^2 samples as an input.

If there is no transmitted signal in the received signal, the off-diagonal elements of R_y are 0, so $T_1 = T_2$. In practice, T_1/T_2 is compared to a threshold η whose selection process is detailed in [150].

Another approach consists in computing the eigenvalues of the covariance matrix [151], *e.g* through a singular value decomposition. In this approach, the test statistics is the maximum eigenvalue to minimum eigenvalue ratio, which is compared against a pre-defined threshold to decide $\mathcal{H}_0$ or $\mathcal{H}_1$:

$$\lambda_{\max}/\lambda_{\min} \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \eta. \tag{1.66}$$

If there is only noise, the correlations are small and the eigenvalues are close to one another, resulting in a small ratio: $\mathcal{H}_0$ is decided. On the contrary, a high ratio means that there are a lot of correlations in the received signal: primary user signal is present, which leads to the decision $\mathcal{H}_1$.

Other test statistics based on the eigenvalues of the sample covariance matrix can be used [152, 153]. For example, the left-hand side quantity in (1.66) can be replaced by $\lambda_{\max}/\Lambda$, where Λ is the average of the eigenvalues of the covariance matrix, or by $\lambda_{\max}/\mathrm{Tr}(\mathrm{cov}(R_y))$, where $\mathrm{Tr}(\mathrm{cov}(R_y))$ is the trace of the covariance matrix R_y.

CovD is particularly adapted to samples acquired through random sampling matrices, and random matrix theory has also relied on CovD to perform blind SS for DSA [150, 151, 153].

Advantages and Drawbacks CovD methods are blind and do not require any knowledge about the primary user signal or the noise. Furthermore, they are more robust to noise level uncertainty than ED methods. However, the benefits of CovD methods come at a price: they require a lot of samples [154] and are computationally complex, especially when eigenvalue decomposition is involved.

1.4.3 Wideband Spectrum Sensing

The continuous increase in data rates ever since the introduction of wireless communications has led to a need for ever higher bandwidths. As a consequence, secondary users that wish to perform DSA need to sense wide frequency ranges in order to find the best communications opportunities. In an infrastructure-oriented scenario, performing SS on wide frequency ranges can also induce an increase in performance compared to NB-SS, in the sense that a wider range of users can have their communications gathered successfully.

A variety of methods for Wideband SS (WB-SS) have been proposed [137]. Early approaches involve splitting, or slotting, a wide frequency band into several narrowband channels, then to perform NB-SS on each of these NB channels, either sequentially (at the expense of increased sensing time) or simultaneously (with the hardware constraints related to Nyquist-rate wideband sampling). Alternatively, WB-SS can be carried out using CS techniques, requiring fewer samples than conventional Nyquist-rate WB-SS.

1.4.3.1 Nyquist Wideband Spectrum Sensing

Nyquist-rate WB-SS techniques include wavelet detection (WD) [155], multi-band joint detection [156] and filter bank based sensing [157]. A common feature of these techniques is that they rely on conventional ADCs operating at the Nyquist rate to sample the wideband signal.

Wavelet Detection When there are adjacent frequency channels with different occupancy statuses (*e.g.* a vacant channel located next to an occupied one), the power spectrum density (PSD) exhibits discontinuities at the edge of the two adjacent channels. As a consequence, frequency channel location estimation (necessary in WB-SS) can be considered as an edge detection problem, for which the wavelet transform is adequate [158].

In WD [155], the PSD of the sampled signal is computed, and the wavelet transform is applied. Local maxima then yield important information about frequency channel locations. Once potential frequency channels are estimated, a NB-SS technique (usually ED) is sequentially carried out in said channels, resulting in a decision on signal-of-interest presence in each frequency channel candidate.

Multi-Band Joint Detection Multi-band joint detection proposes to conduct ED in every NB subband simultaneously rather than sequentially. For this purpose, the WB is sliced into K adjacent non-overlapping NB subbands. For the ith NB subband ($1 \leq i \leq K$), as per conventional ED description, a threshold γ_i has to be selected, based on the target probabilities of detection and false alarm. The authors of [156] propose to perform a joint optimization on a threshold vector $\gamma = [\gamma_1 \dots \gamma_K]^T$. The threshold vector is then used to carry out ED in parallel over all NB subbands.

Filter Bank Based Sensing Similar to multi-band joint detection, filter bank based sensing aims to perform ED simultaneously in adjacent NB subbands. For this purpose, this technique relies on implementing band-pass filters through a poly-phase decomposition of the prototype filter [157]. Filter bank analysis consists in an array of band-pass filters. The input of each band-pass filter is the received signal, and its output is a single NB subband, on which ED is carried out to decide $\mathcal{H}_1$ or $\mathcal{H}_0$.

1.4.3.2 Sub-Nyquist Wideband Spectrum Sensing

The CS framework provides powerful tools for a variety of applications (see Section 1.3.1 and surveys referenced therein). In particular, WB-SS can particularly benefit from CS principles and tools. A definitive requirement for application of CS to WB-SS is that the signal of interest must be sparse in some domain (usually the frequency domain, although alternative domains can be considered). Two approaches for applying CS tools to WB-SS have been favored: partial USLE resolution and adaptation of Nyquist-rate SS techniques to CS.

Partial USLE resolution Solving the CS USLE (1.15) using either convex optimization, greedy algorithms or other methods, typically involves estimating the location of the non-zeros (that is, the frequency support of the signal) as an intermediary step to full signal recovery. As mentioned in Section 1.3.1.1, this intermediary step is an opportunity to perform WB-SS, as the outcome of SS is generally the frequency support of the signal of interest, or characteristics derived from it. An advantage is that this approach is compatible with the different hardware architectures and their relative USLEs/sampling matrixes. However, this approach

has a somewhat high complexity [159], although complexity depends on the method used for USLE resolution.

Adapting Nyquist-rate SS techniques to CS A path towards compressed WB-SS that has attracted attention is the adaptation of SS techniques to compressed samples. More precisely, it is to note that the techniques presented in Section 1.4.2 all compute a statistic from the Nyquist-rate samples, then compare it to some threshold. For a given method, it is often possible to compute the same statistic from the compressed samples directly, without recovering the signal of interest or its frequency-domain support first.

Examples for various SS methods include the following. An early example is the development of a compressive wavelet-based edge detector [24]. The smashed filter, a compressive version of the matched filter, was introduced in [160]. Compressive energy detection was proposed in [140, 161], and a power spectrum computation from compressed samples is discussed in [162]. Cyclostationary detection from compressed samples is explored in [163, 164]. A Bayesian CS approach to WB-SS is put forward in [165]. Other examples of the computation of signal-induced characteristics from compressed samples are referenced in [159].

An observation about these proposed techniques is that they are often coupled to specific constructions of the CS sampling matrix A in the CS USLE $y = Ax$. These constructions are sometimes compatible with a limited number of hardware architectures, *e.g.* the MWC or other RD-based schemes, and sometimes limited to theoretical objects that have not been, to the best of our knowledge, the subject of any hardware implementation in communications yet, *e.g.* random matrices. As a consequence, these techniques rely on assumptions about the properties of the sampling matrix A that cannot be met in general. This specificity hinders their widespread use.

1.4.4 Conclusion on Spectrum Sensing

In this section, we first reviewed the main techniques for NB-SS, a flavor of SS which is mostly dedicated to the detection of a signal in a narrow band. Afterwards, we presented WB-SS, which aims at detecting the support of the signal of interest in a wide band - possibly with a somewhat coarse granularity. WB-SS can be performed either by applying NB-SS techniques to narrowband subbands of the wide band, either sequentially or simultaneously, or by using principles borrowed to CS. Although WB-SS has been the focus of quite some research for the past several years, much of it remains to be explored.

In Chapter 2, we will contribute to WB-SS; yet we will apply SS techniques as a preprocessing step for the resolution of the USLE of CS rather than use CS to solve a SS-related problem.

1.5 Conclusion

This chapter served several purposes. First, it provided elements to grasp the fundamentals of CS and SS. Second, it reviewed popular and recent techniques in CS and SS. Third, it lay the

ground for the presentation of our contributions to both fields, keeping in mind the objective of developing a prototype of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure.

Glancing at CS and SS, the uninitiated eye could see subjects that are both well-investigated and covered by proficient literature. While these subjects are certainly not *terræ incognitæ*, there is still leverage for interesting discoveries in both fields: as the sphere of our knowledge gets bigger, so does our interface with the unknown. Although the era of major theoretical breakthroughs in CS seems to be mostly behind us, many practical aspects remain to be improved upon, for the industry to largely embrace CS.

In the remainder of this manuscript, we will build upon what we have reviewed in this chapter, and we will fill the gaps between what the literature offers and what we need for our envisioned prototype. More precisely, in Chapter 2, we will provide contributions in SS regarding the estimations of i) unknown noise variance for ED and ii) spectral support of signals of interest. In Chapter 3, our contribution to CS will be related to an essential setting of the MRS CS scheme. Chapter 4 will be a case study of a sampling infrastructure prototype, aimed at empirically validating our contributions through realistic scenarios.

Chapter 2

Spectrum Sensing: Noise Variance and Support Estimations of Sparse Signals

2.1 Introduction

Success or failure of a sampling infrastructure depends heavily on its ability to identify signals of interest (SoIs) and their location in the time and frequency domains. While Spectrum Sensing (SS) literature is proficient, we have found that some specific signal processing steps have received less coverage than others, and are skipped as trivial or taken for granted in many publications. In particular, there are two steps that take place in our MRS-based sampling infrastructure prototype, but that have not been, to the best of our knowledge, the subject of satisfactory, implementation-proof research. These steps are the blind estimation of the noise variance in an AWGN channel and the estimation of the frequency-domain support of a sparse SoI. Consequently, we have developed a novel method for each of these two problems. The noise variance estimator has been the subject of a peer-reviewed publication in an international conference [3] and of a patent submitted to INPI [4], the French patent governing body. The support estimator has been the subject of a publication, currently in the submission process.

2.1.1 Motivation for estimating the noise variance and the signal support

For our envisioned sampling infrastructure presented in Section 0.2.3 of the general introduction, we wish to identify and store sub-Nyquist samples of any SoI present. As we consider wideband sparse signals, the SoI can be multi-band, meaning that the possible narrowband frequency-domain components of the SoI are not necessarily adjacent. Naturally, we wish to store samples of the SoI components only, we are not interested in keeping samples corresponding to white noise. It is therefore of utmost importance that we perform WB-SS, that is, that we identify the support of the SoI. Once the support of the SoI is estimated, we can apply the reduction procedure promoted by the authors of the MRS scheme [20], and presented in Section 1.3.2.2 of Chapter 1 of this manuscript. Indeed, this reduction procedure allows to solve the MRS-related

USLE more easily (we still need a procedure to recover Nyquist-rate samples from compressed samples, that we have acquired in a sub-Nyquist fashion). Note that the authors of [20] did not, however, provide a way to estimate the support of the SoI: filling this gap is the main motivation of the present chapter.

In a way, when some solve (sometimes partially) a CS-acquired USLE to extract the support of the SoI ([166, 167, 168]), we reverse the order of the operations. We use non-CS-related techniques (presented here) to obtain the support of the SoI, and then use this acquired information about the SoI to solve the USLE more easily.

Remember that the MRS relies on undersampling, which means that the samples acquired at each branch exhibit aliases of the same SoI frequency components. These aliases contain the same information as the original SoI components, but from one branch to another, they are in different locations of the spectrum, because the signal was not sampled at the same rate. Using our method, we will first be able to estimate the support of the SoI in each branch of the MRS. We will then aggregate these so-called partial support estimates, using the method described in [20], to form an estimate of the support corresponding to Nyquist-rate sampling. Afterwards, the reduction procedure is applied.

Our method is based on sample-wise energy detection (ED). Instead of applying energy detection (a NB-SS technique) to a large number of samples, we apply it to one sample, or at most to a very limited number of samples. This is done in an effort to increase the resolution of our estimator as much as possible.

Understandably, our method shares some characteristics with ED applied as a NB-SS technique. In particular, knowledge of the background noise variance is crucial [138, 169, 170, 171, 172, 173], because it is used to set the threshold η discriminating a SoI component from noise. However, the existing methods for blind noise variance estimation were not in line with the specificities of our use case, for several reasons that are exposed in Section 2.2.1.1. As a consequence, we developed our own method. It is based on the separation of noise-only values and signal-plus-noise values in the frequency representation of the received signal. This separation is conducted using the K-means algorithm and requires for the signal to exhibit some sparsity and compacity. Note that noise variance estimation is important for other signal processing applications, such as SNR estimation too [174, 154, 141]: the scope of our novel method is broader than what we intend to use it for.

2.1.2 Chapter Outline

In this chapter, we put forward two methods related to SS of sparse signals: noise variance estimation and support estimation. We will start by describing our noise variance estimator, because its output is used as an input to our support estimator. Section 2.2 will be dedicated to the noise variance estimator, while the support estimator is presented in Section 2.3.

2.2 Noise Variance Estimation

2.2.1 Introduction to Noise Variance Estimation

In the Additive White Gaussian Noise (AWGN) channel model, on which this section focuses, the variance σ_w^2 of the AWGN is the only variable at stake, meaning that acquiring channel state information (CSI) consists in finding σ_w^2. Here, we will therefore aim at estimating the variance σ_w^2 of the AWGN in the propagation channel, in the presence of an unknown signal: this constitutes blind estimation.

2.2.1.1 Previous works

A common solution to acquire CSI is for the sender and receiver to rely on protocol-specific pilots, located in the overhead of the transmissions. However, this may be impratical or impossible if a sender and receiver use two different, mutually unintelligible protocols, which is often the case in Detection Theory applications [138]. An alternative is to listen to the channel at a time when it is free of transmissions [141]. Yet, this method is unsatisfactory because it can be difficult to determine whether this channel is free of transmissions or not, without having information about the channel.

When some information about the signal is known, data-aided estimators can be implemented. Pauluzzi et al [174] compared several SNR estimators that work only on signals of certain modulations, or need knowledge of at least some parameters of the transmitted signals. Mathew et al [170] estimated the AWGN variance by applying autocorrelation on the cyclic prefix of LDACS-modulated signals. These methods are not suited for blind AWGN variance/SNR estimation.

For blind AWGN variance/SNR estimation, a popular approach ([169, 154, 171]) involved the computation of the eigenvalues of the sample covariance matrix and their classification into either noise or signal values using various criteria. While effective, these methods have a high complexity, of at least $\mathcal{O}(N^3)$ (the problem size N is the number of samples), can be difficult to implement and require a high number of samples (typically $N \geq 40,000$).

Another technique for AWGN variance estimation was proposed by Makovoz in [2]. The author considered time-sparse signals, so that a sample vector consists mostly of AWGN. Noise realizations follow a Gaussian distribution and SoI realizations are considered as outliers from the aforementioned Gaussian distribution. The method consists first in finding a subset of samples without SoI outliers, then in sorting this subset in order to recover a part of the cumulative distribution function of the normal distribution whose variance is that of the Gaussian noise. While this method has a high accuracy, its precision is rather poor: in many scenarii, the results have a standard deviation of about 20% around the true value of the variance. A possible cause for this limited precision is the slow convergence between the statistics of the realizations of a random variable following a given distribution and the parameters of the distribution itself: the first converges only asymptotically towards the second. Statistics of realizations of a random variable too far apart from the parameters of a distribution lead to inadequate numerical parameters for equation resolution, resulting in turn in a poor precision. Further, this method

is limited to SoIs occupying less than 50% of the total bandwidth. In our proposed sampling infrastructure, we never consider SoIs with an occupancy ratio higher than 50% when they are sampled at the Nyquist rate; however, when undersampled in an MRS branch, spectrum folding causes signal components to be aliased over spectrum holes, leading to an increase of the occupancy ratio (more details of this process can be found in Chapter 4). As a consequence, we are interested to consider the performance of a noise variance estimator for signals that have an occupancy ratio greater than 50%.

2.2.1.2 Principle and characteristics of our proposed method

We propose a new method to determine the variance σ_w^2 of an AWGN. Our method, conceptually close to that of Makovoz, relies on separating frequency-domain samples containing noise only from samples containing a mixture of SoI and noise. This separation is mainly conducted using K-means clustering [175], applied to preprocessed data and followed by a postprocessing step. Our method has a high accuracy (non-biasedness) and a high precision (low standard deviation), yet a low complexity, and requires a limited amount of signal samples to achieve satisfactory accuracy and precision. While the considered signal is unknown, it should follow two hypotheses: a certain degree of sparsity, meaning that the signal frequency support should not occupy the entire considered bandwidth, and some compacity, meaning that the signal support elements should be grouped enough and allow for noise-only sub-bandwidths. Sparse multi-band signals, which stem from multiple access in communications, are a class of signals for which this method is particularly adapted. Note that while we focus on frequency-sparse signals, this method is actually applicable to any sparsity.

2.2.2 System Model

Notations: for a given baseband signal x, let $x(t)$ be its continuous time-domain representation, $x[n]$ be its discrete time-domain representation (sample), $x = [x[0]x[1]\dots x[N-1]]^T$ be the vector of signal samples of length N, and X be the discrete Fourier transform (DFT) of x, multiplied by $1/N$ for normalization (the elements of X, are denoted $X[k]$, $0 \leq k \leq N-1$).

Let $x(t)$ be a frequency-domain sparse, noiseless and continuous baseband signal. Its support is the set of frequencies where X is non-zero. Signal x propagates through the AWGN channel. In the frequency domain, the received sampled signal is of form

$$Y = X + W, \tag{2.1}$$

where $W[i] \sim \mathcal{N}(0, \frac{1}{2}\sigma_w^2) + j\mathcal{N}(0, \frac{1}{2}\sigma_w^2)$, where $0 \leq i \leq N-1$ and where σ_w^2 is the unknown AWGN variance.

Indices of frequency samples present (resp. absent) in the support of X correspond to Y's "signal and noise mixture" (resp. "noise only") elements. Fig. 2.2 displays the energy of the noiseless signal X, noisy signal Y and the support of X (black intervals). At acquisition, the support of X is unknown.

While signal x is unknown, it is subject to two hypotheses:

1. **Sparsity** – The support of X should occupy a limited portion of the total bandwidth. Were it not the case, there would not be a sufficient quantity of noise-only frequency samples for our method to operate successfully. The sparsity of X is measured by its occupancy ratio, which is the number of samples in the support of X divided by N, the total number of samples. The sparsity requirement depends on the size N of the sampling vector and is quite lax. For example, for a sample vector of size $N = 5000$ samples, noise variance estimation can be carried out with an occupancy ratio of 90%. Other suitable occupancy ratios are presented in Section 2.2.4.

2. **Compacity** – The support of X can spread out over the entire bandwidth, but should always contain uninterrupted, unoccupied spans of the bandwidth (also called gaps). This allows for a smoother preprocessing step. The gaps can be fairly narrow (less than 2% of the bandwidth is typically enough for a sample vector of size $N = 5000$ samples, as shown in Section 2.2.4).

Throughout the entire method, we will focus on the "SoI-plus-noise" mixture Y.

2.2.3 Noise Variance Estimation based on K-means clustering

The general principle of the method is to separate samples of Y corresponding to noise only (located outside of the support of X) from those corresponding to a mixture of noise and signal (located inside the support of X), using K-means clustering, a partitioning method. This separation is conducted on the energy of each sample in the frequency domain.

Fig. 2.1 summarizes the important steps of our method, which will be described in this section.

A guiding example To illustrate the successive steps of our method, an example based on a given signal is provided throughout this section. The steps of our method are exemplified in Figs. 2.2, 2.3, 2.4 and 2.5. The signal used as an example for these figures consists in two filtered QPSK sub-signals, located at two different positions in the frequency spectrum. The characteristics of the sub-signals are as follows. Each sub-signal occupies a fraction of the entire considered bandwidth, so the total occupancy ratio is $\rho = 30.2\%$. An AWGN is added to the entire bandwidth; overall, the SNR is 2.1 dB for the simulations in this section.

This example is intended for the step-by-step illustration of the method. Section 2.2.4 will provide simulations results with various SNRs.

Fig. 2.2 displays the noiseless signal X and noisy signal Y. Having access only to Y, the aim is to isolate noise-only samples (samples outside of the black intervals) using our novel clustering-based method.

2.2.3.1 Preprocessing: regrouping noise values closer together

The first step of the method starts with a moving average smoothing step. The energy of the vector Y (the element-wise squared modulus of Y, divided by its length) is windowed (in the

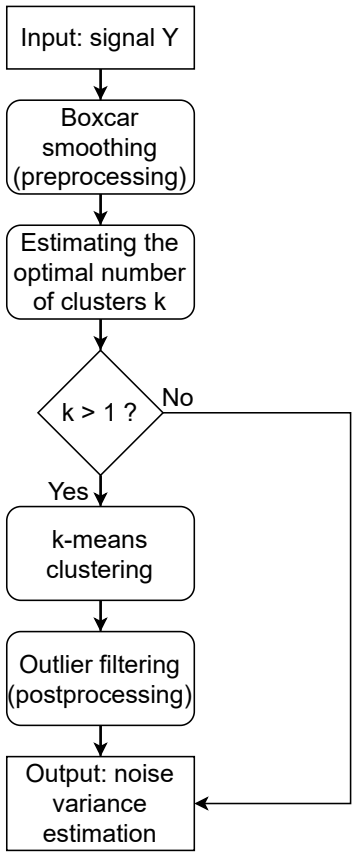


Figure 2.1: Description of our K-means-based method for blind noise variance estimation.

frequency domain) by a rectangular window of length w_l and of height $1/w_l$ (see Fig. 2.3). This operation yields a new vector, P, defined as follows:

$$P[i] = \frac{1}{w_l} \sum_{m=i}^{i+w_l-1} |Y[m]|^2, 0 \leq i \leq N - w_l. \tag{2.2}$$

The aim of this $\mathcal{O}(N)$-complexity preprocessing step is the reduction of the scatter of the values of $|Y|^2$ corresponding to noise only. Fig. 2.3 displays the signal energy $|Y|^2$ (before preprocessing) and P (after preprocessing). We can see that after the vector is preprocessed, the scatter between noise-only values is considerably reduced[1].

Indeed, the values of $|Y[k]|^2$ follow a χ^2 distribution with 2 degrees of freedom (d.o.f.), while the values of $P[i]$ follow a χ^2 distribution with $2w_l$ d.o.f., multiplied by $1/w_l$. The high quantiles of the $\chi^2(2w_l)$-based distribution are lower than those of the $\chi^2(2)$ distribution, hence the values of P in the noise-only domain are less spread out than those of Y. The same reasoning goes for the "SoI-and-noise mixture" part of Y, in the way that noise incurs less value dispersion in P than in Y.

However, the size of the window w_l is limited by side effects that take place at the edges of the support of X. Indeed, windowing implies that some values of P contain a mixture of noise-only components (those outside of the support of X) and of SoI-plus-noise-mixture components (those inside the support of X). While the impact of such elements is limited for small-sized

[1]This is done at the expense of the frequency resolution: an error in support estimation is deliberately added.

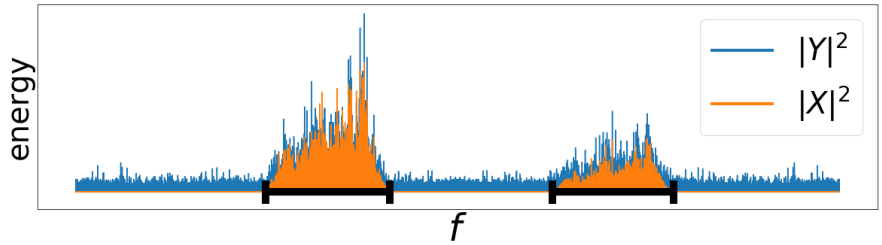


Figure 2.2: Energy of noiseless signal X, its support (black intervals), and noisy signal $Y = X + W$. Signals are represented in the linear scale.

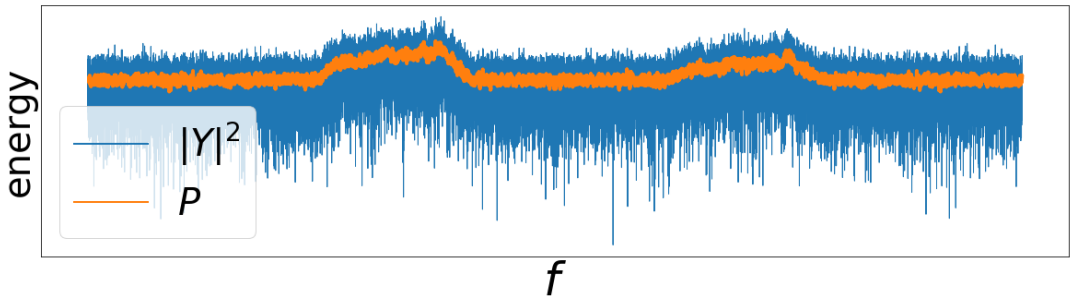


Figure 2.3: Before and after the preprocessing step. Signals are represented in the log scale for better visibility. The window length is $w_l = 30$ samples.

boxcar windows, excessively high values of w_l can alter the efficiency of the subsequent clustering step.

Another characteristic that can detrimentally impact the performance of the method is the violation of the compacity hypothesis on X. Indeed, the absence of large enough gaps in the support of X result in a lack of "pure" noise-only values in P, in which case the method fails (meaning that the noise variance will be overestimated).

The trade-off between lower noise value scatter and side effects is discussed in Section 2.2.4.3.

2.2.3.2 Estimating the optimal number of clusters

After preprocessing, the values in P are ready to be classified into categories, or clusters. K-means clustering (described more thoroughly in the next subsection) does not detect by itself the optimal number of clusters (ONC) to partition the data. As a consequence, the ONC is an input to K-means clustering, and needs to be determined beforehand.

At a first glance, the most relevant partitioning for values in P would be between a "noise only" set and a "SoI plus noise" set (and potentially a "mixture of noise-only and SoI plus noise values" set). However, the absence of any particular assumption on the values of the SoI X renders the partition in two (or three) clusters possibly less relevant than another partition: what if SoI X is the sum of, say, five sub-signals of different, constant amplitude? what if there is no signal (meaning the support of X is empty)? In the first case, the ONC could be six (five clusters for SoI components and one for noise-only components), while in the second case, the ONC could be one. Consequently, making a guess on the ONC is hasardous.

To estimate the ONC, we suggest to use the average silhouette width [176]. For a given partition of values into clusters, let value i be associated to cluster C_i. Let

$$a_i = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j) \tag{2.3}$$

be the mean distance between i and the other values in C_i (in our case, $d(.,.)$ is the 1D Euclidean distance).

Now let

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \tag{2.4}$$

be the smallest distance of i to all points in any other cluster than C_i.

The silhouette of i is defined as follows:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}. \tag{2.5}$$

Finally, the average silhouette width is defined as the average of the silhouette $s(i)$ over all values i in the entire dataset (here, the vector P).

One way to determine the ONC is to exhibit a partition using K-means clustering for different numbers of clusters (this number usually ranges from 2 to 7), and to compute the average silhouette width for the given partition.

The average silhouette width ranges from -1 to 1 and measures the quality of the partition of a dataset into clusters (the higher the average silhouette width, the more relevant the partition). A partition is usually considered relevant if its corresponding average silhouette width is above 0.7 [176]. The number of clusters k for which the corresponding average silhouette width is maximal becomes the ONC.

Fig. 2.4 displays the average silhouette width for different number of clusters, computed by averaging the silhouettes computed on our example signal using (2.3), (2.4) and (2.5). All scores are above 0.7. This is an indication that clustering is relevant. Here, the ONC k is 3.

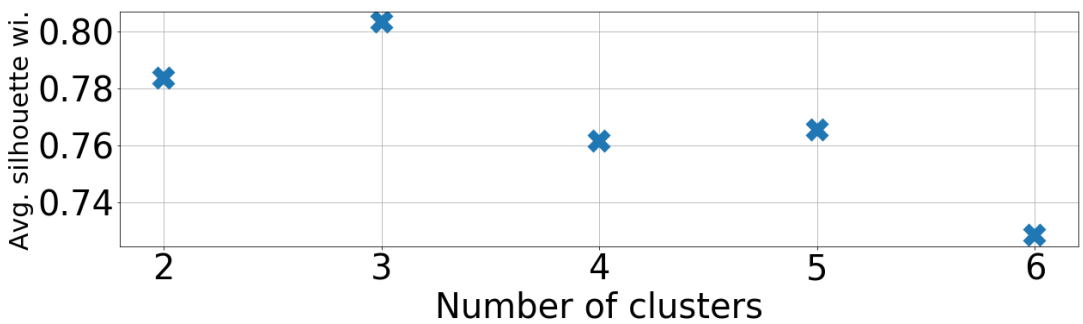


Figure 2.4: Average silhouette widths for different number of clusters.

We now present two refinements to the ONC estimation.

First, if the ONC k determined during this step has an average silhouette width below 0.7, we consider that clustering is inappropriate, and no partitioning is conducted. k is then set to 1 and we skip to the last step of the estimator, depicted in Fig. 2.1 and presented in Section 2.2.3.5.

Empirically, this happens in two situations: a) when there is no SoI (support of X is empty); partitioning into 2 or more clusters is therefore useless and even detrimental to correct noise variance estimation, or b) the signal of interest is spread out on too many frequency samples (in violation of the parcimony and/or compacity hypotheses), and no partition whatsoever allows the exhibition of a cluster containing noise-only values. In case a), the method is resilient to absence of SoI and typically succeeds, (*i.e.* the noise variance estimation is correct, while in case b), the method fails. A drawback of the method is that during its processing (and even afterwards), it is difficult to identify whether we are in the first or in the second situation.

Second, while the algorithm used for K-means clustering has a low complexity, the computation of the average silhouette width is fairly resource-intensive because it is more complex. This problem can be mitigated by computing the average silhouette width on a subset of values of P: empirically, a random draw (without replacement) of 100 values of P has been shown to greatly reduce the resource use of this step of the method without any significant negative impact on the ONC estimation. As a consequence, the complexity of this step is constant: $O(1)$.

2.2.3.3 K-means clustering

If $k \geq 2$, the Lloyd's algorithm for K-means clustering [177] is applied to all values of P. The K-means algorithm is iterative. During initialization, k centroids are determined using the k-means++ method [178]. At each iteration, each value of the vector P is assigned to the closest centroid; the set of values assigned to a given centroid constitutes a cluster. The position of the centroid is then updated and set to the average of data points in the cluster, and the iteration is completed. The algorithm stops when the position of centroids remains unchanged from one iteration to the next, or when the maximum number of iterations is reached (this condition is implemented so as to make sure the algorithm ends).

The cluster with the lowest mean, or minimal mean cluster (MMC), corresponds mostly to values of P containing only noise, while other clusters contain values of P corresponding to the SoI-plus-noise mixture.

The values in the MMC are distributed according to a $\chi^2(2w_l)$ distribution, already weighted during the preprocessing step so that its expected value is $\hat{\sigma}_w^2$, that is, the estimate for the noise variance. As a consequence, a first estimation $\hat{\sigma}_w^2$ of the noise variance σ_w^2 is the arithmetic mean of the MMC. Due to its high number of d.o.f., the $\chi^2(2w_l)$ distribution converges towards a normal distribution $G \sim \mathcal{N}(\sigma_w^2, \sigma^2)$.

The complexity of this step is that of the Lloyd's algorithm, which is $O(kNi)$ [179]. i is the number of iterations, capped to a maximum value in our implementation. Since $i, k \ll N$, we consider this step to be of linear complexity.

2.2.3.4 Postprocessing step: filtering the outliers

The MMC obtained in the partitioning step of the method mostly consists in noise-only values, but not only. Indeed, values corresponding to low-energy SoI, or values containing a noise and SoI-plus-noise mixture (due to the side effects of the windowing, occurring at the preprocessing step), can be closer to the centroid of the MMC than to centroids of other clusters. This

introduces a bias that causes the MMC to diverge from a collection of points that would be obtained through a Gaussian process (see the tail in Fig. 2.5). To mitigate this bias, we consider that these aforementioned, non-noise-only values are outliers for G. This is justified by the fact that if the previous clustering step was successful, these values are typically much higher than noise-only values and present in small amounts, therefore easy to single out.

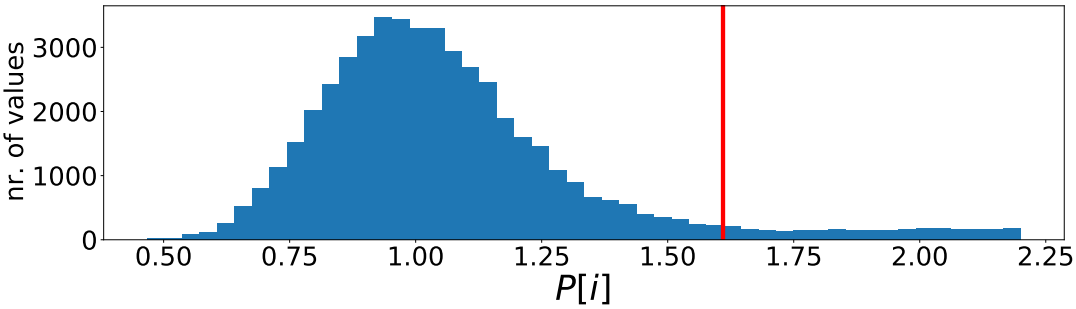


Figure 2.5: Histogram of the values in the MMC before postprocessing, based on our example signal. The values that remain after the postprocessing step are located on the left-hand side of the vertical line.

During the $\mathcal{O}(N)$-complexity postprocessing step, such outliers are withdrawn from the MMC using the three-sigma rule [180]. As 99.7% of values drawn according to G are located between $\sigma_w^2 - 3\sigma$ and $\sigma_w^2 + 3\sigma$, values of the MMC above $\sigma_w^2 + 3\sigma$ most likely do not correspond to realizations of G. Therefore, this step consists in computing the mean μ and standard deviation s of the values in the MMC and in removing values above $\mu + 3s$ from the MMC. This process is repeated until no value is removed, the MMC is then considered consistent. Convergence is typically reached between 1 and 10 iterations; nonetheless, a limit of 20 iterations is set, in order to guarantee termination.

Fig. 2.5 displays a histogram of the values in the MMC before postprocessing. The vertical line is located at the highest value remaining after applying the three-sigma rule iteratively on the MMC. Therefore, it represents the limit between values remaining in the MMC (left) and removed values (right) after the postprocessing step.

2.2.3.5 Output

After these different steps, the MMC is considered to contain exactly the noise-only values of P. Its mean is computed and returned: it is our noise variance estimation $\hat{\sigma}_w^2$. Our estimator is noted $KNVE$. Note that if k was set to 1 when computing the ONC, clustering is not performed, all the values of P are considered as noise, and the mean of the values of P is computed and returned.

2.2.4 Simulation Results and Discussions

2.2.4.1 Comparison with previous works

First, we compare the $KNVE$ to Makovoz' MSL and $AMSL$ [2]. For this purpose, we generate sets of SoI-only (noiseless) data and noise-only data. The noise-only data are normally distributed, with a mean of zero and a standard deviation of 1. The SoI-only data consists in data points drawn either uniformly between 3 and 8 or normally with a mean of 4 and a standard deviation of 1. These simulation hypotheses are those of [2]. In order to meet the compacity hypothesis on the structure of the signal, we split the bandwidth into 10 channels in which the SoI-only data are slotted (this constraint is not present in [2]). For the sparsity hypothesis, we limit the occupancy ratio to a maximum value of 90%, higher than [2]'s 50% fraction of outliers. Finally, the simulations are run for X vector lengths of $N = 300$, 1000 and 5000 samples. Each simulation being repeated 1000 times so as to compute the mean and the standard deviation of the MMC estimator. The results are displayed in Table 2.1.

		Spectrum occupancy ratio of signal-of-interest-only data					
SoI Type	Size	20%	30%	40%	50%	70%	90%
Uniform	100	*.98* (*.28*)	*1.06* (*.34*)	*1.09* (*.38*)	*1.08* (*.50*)	-	-
	300	1.00 (.10)	1.92 (2.30)	4.91 (5.32)	8.46 (6.74)	11.2 (6.87)	11.5 (2.33)
	1000	.997 *1.01* (.051 *.20*)	.995 *1.03* (.057 *.21*)	1.000 *1.01* (.061 *.18*)	1.000 *.99* (.102 *.09*)	1.316 (.836)	2.783 (1.319)
	5000	.993 (.023)	.993 (.024)	.995 (.026)	.994 (.029)	.996 (.038)	1.001 (.065)
Gaussian	100	*.98* (*.28*)	*1.06* (*.34*)	*1.11* (*.40*)	-	-	-
	300	1.01 (.10)	1.63 (1.27)	3.16 (2.73)	5.08 (3.44)	6.43 (3.69)	6.35 (1.24)
	1000	.996 *1.02* (.054 *.19*)	1.000 *1.03* (.056 *.21*)	.997 *1.01* (.063 *.19*)	1.006 (.080)	1.192 (.472)	1.930 (.677)
	5000	.995 (.023)	.995 (.025)	.995 (.026)	.994 (.029)	.995 (.037)	.999 (.067)

Table 2.1: The mean (standard deviation) of the proposed noise variance estimator ([2]'s $AMSL$ in italics). The smoothing window length is $w_l = 30$ samples.

For a limited amount of samples ($N = 300$ samples), our estimator performs poorly, and far worse than the $AMSL$ estimator of [2] with 100 samples[2]: these failures are typically caused by the side effects induced by the preprocessing step. However, with more data ($N = 1000$),

[2]For $N = 100$ samples, whatever the signal parameters, our method almost never identifies more than one cluster. Consequently, the method is ineffective, so we chose not to include them in Table 2.1.

the $KNVE$ outperforms the $AMSL$ for almost every occupancy ratio, both in accuracy (the closer to 1 the mean, the more accurate) and in precision (the closer the standard deviation, or scatter, to 0, the more precise). Estimating on even more values ($N = 5000$) lowers the estimation scatter and, perhaps more importantly, allows to estimate the noise variance in datasets with more SoI-plus-noise-mixture datapoints than noise-only datapoints, thus going beyond [2].

2.2.4.2 Robustness to low SNRs

Now, we test our method for different SNRs. The signal is generated as follows. As in Section 2.2.4.1, the bandwidth is split into 10 channels. A subset of these channels, whose size is proportional to the given occupancy ratio, is randomly selected and set to be the frequency support of the noiseless signal X. In this support, a noiseless signal is generated in the following manner. Each frequency sample is set to a value drawn according to the so-called "constant energy" distribution defined as:

$$X[i] = \sqrt{E}e^{jU}, \tag{2.6}$$

where E is the energy per sample of X, $U \sim \mathcal{U}(0, 2\pi)$ and $\mathcal{U}$ is the uniform distribution. Inside its frequency support, each sample of X is of constant energy E. Outside of the support, $X[i] = 0$. Let C_X be the number of elements inside the support of X.

Noise-only values $W[i], 0 \leq i \leq N-1$ are now generated according to the distribution presented in Section 2.2.2. Finally, the noiseless data vector X and the noise vector W are added to give Y.

The SNR varies with E, the energy in each sample of the support. The relationship between the linear SNR, E, σ_w^2 and the occupancy ratio ρ is the following:

$$\text{SNR}_{\text{linear}} = \frac{C_X E}{\sigma_w^2 N} = \rho \frac{E}{\sigma_w^2}. \tag{2.7}$$

In this experiment, the size of Y is set to $N = 5000$ samples, the window length to $w_l = 30$ samples, and the occupancy ratios vary from 20% to 90%. The SNR (dB) varies from -15 dB to 7 dB. Each simulation is averaged over 100 runs.

Fig. 2.6 displays the mean $\hat{\sigma}_w^2$, the noise variance estimated by our method, versus SNR (dB), for various occupancy ratios. As a reminder, in our case, the actual noise variance is $\sigma_w^2 = 1$. The first observation is that for each occupancy ratio ρ, there are three regimes: a "low SNR" regime, in which the estimator basically picks the entire energy of the SoI-plus-noise mixture as noise energy; a "high SNR" regime, in which only the noise values are detected as such; and a transition regime in between. Let a "success" (resp. "failure") estimation be an estimation with a relative error under (resp. above) 5%. In the "low SNR" regime, the success rate of the estimator is low; in the "high SNR regime", its failure rate is low; finally, in the intermediary regime, both failure and success occur frequently. As a consequence, the estimator is only reliable in the "high SNR" regime.

We can also observe that the entrance in the "high SNR" regime depends on the occupancy ratio. In particular, the edges of the different SNR regimes vary for different occupancy ratios:

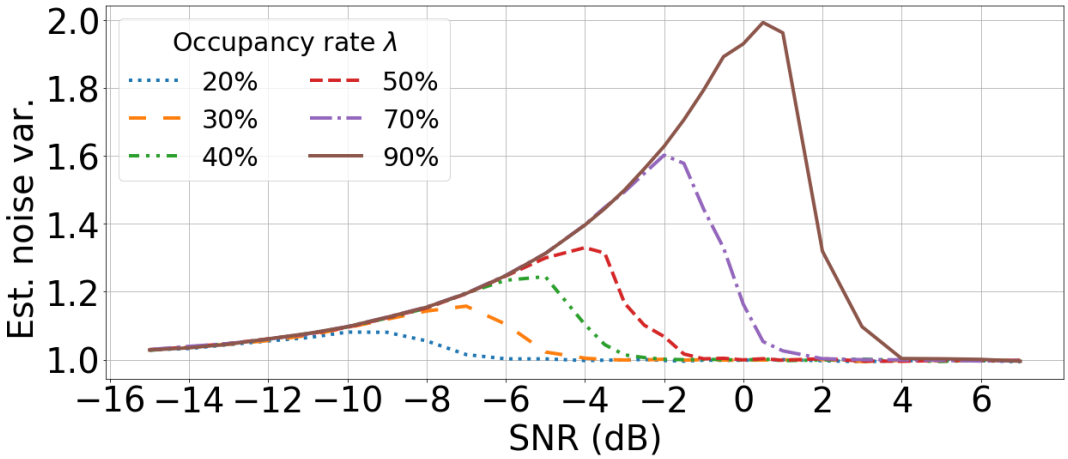


Figure 2.6: Noise variance estimation $\hat{\sigma}_w^2$ vs SNR (dB) for various occupancy ratios ρ. $N = 5000$ samples, $w_l = 30$ samples.

for $\rho = 20\%$, the transitions are at -11 dB (low SNR regime/intermediary regime) and -7 dB (intermediary regime/high SNR regime), while for $\rho = 90\%$, these transitions are respectively at 0 dB and 4 dB. Two factors contribute to this phenomenon. First, for a given signal sample energy E, the total energy increases with the occupancy ratio. However, the values our estimator uses are aggregated from local frequency samples in a bottom-up fashion. To oversimplify, the estimator performance depends on the energy per sample E, not the total energy $C_X E$. Consequently, for a given value of E and a given estimator performance, a higher occupancy ratio results in a higher SNR, but estimator performance is unchanged (as long as side effects mentioned in Section 2.2.3.1 do not take place on a massive scale). Second, the lower the occupancy ratio, the more numerous the noise-only values in Y and in P: this higher number eases the clustering process, an effect already witnessed by the authors of [154].

It is important to note that our proposed noise variance estimator is robust to low SNRs, in particular for sparse SoIs. Even with a very high occupancy ratio of 90% of the total bandwidth, a mere 4 dB SNR ensures a very good noise variance estimation. However, the estimator performance can be improved by the tuning of another parameter: the smoothing window length w_l.

2.2.4.3 Smoothing window length w_l: a trade-off

As mentioned in Section 2.2.3.1, having a high smoothing window length allows to reduce the scatter of noise-only values, but makes the process subject to side effects (namely, the higher proportion of noise/SoI-plus-noise mixtures). On the other hand, a small window length yields fewer side effects but makes it harder to discriminate between SoI-plus-noise and noise-only values during the clustering step.

Fig. 2.7 displays the mean $\hat{\sigma}_w^2$ versus SNR_{dB} for simulation parameters identical to the experiment conducted in Section 2.2.4.2, except for the following parameters: the occupancy ratio is fixed to $\rho = 70\%$ and the window length w_l ranges from 30 samples to 200 samples. For window length $w_l = 100$ samples, the three different regimes for the estimator are shifted to

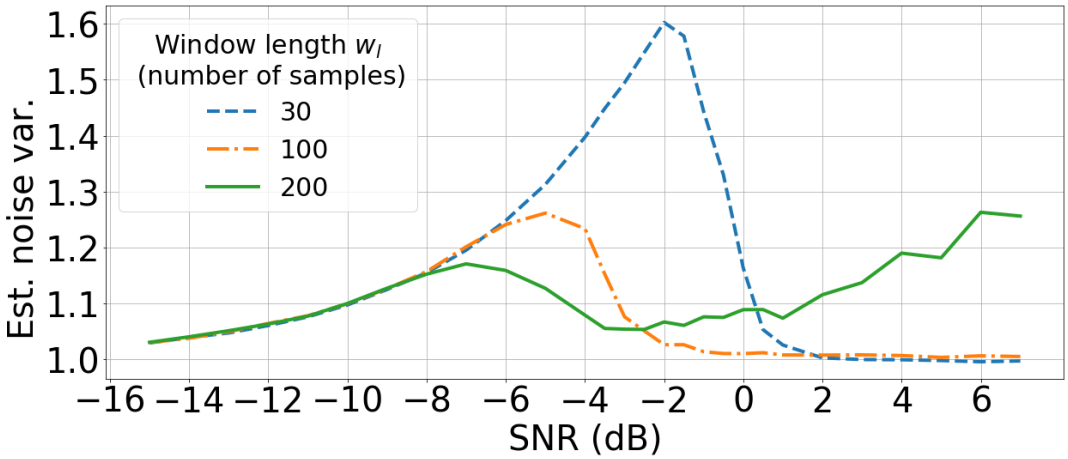


Figure 2.7: Noise variance estimation $\hat{\sigma}_w^2$ vs SNR (dB) for various smoothing window lengths w_l. $N = 5000$ samples, $\rho = 70\%$.

the left, towards the lower SNRs, by approximately 4 dB, compared to window length $w_l = 30$ samples. This is an illustration of the improvement brought by the reduction of the scatter of noise-only values through window smoothing.

However, there is a limit to how much window lengths can be increased, due to the aforementioned side effects. Consider the $w_l = 200$ samples line in Fig. 2.7. After the transition regime, there is no "high SNR" regime in which the estimator has a high success rate. Indeed, because of the side effects at the edges of the support of X, there is not enough noise-only values in P for the estimator to have a high success rate.

To quantify the maximum suitable window length w_l, the compacity of X, or equivalently the size of the gaps in its support (corresponding to noise-only values in Y), is a crucial aspect. If the frequency support of X is divided into channels, then the number of channels in the total bandwidth is an indication of the compacity of X. For example, a 5-channel bandwidth gives a more compact signal than a $1,000$-channel bandwidth. Fig. 2.8 displays the maximum empirical window length w_l, as a percentage of N, for which the noise variance estimation is acceptable (relative mean error under 10%, standard deviation under 10%) versus channel relative width, defined as the inverse of the number of channels in which the total bandwidth is divided. The number of points is $N = 1000$ and the occupancy ratio is 40%. Each simulation is averaged over 50 runs.

From Fig. 2.8, we can see that the higher the relative channel width, the more compact the signal, and the higher the maximum smoothing window size. Another observation is that noise variance estimation on low compacity signals (relative channel width around 2%) can only perform well with windows of maximum size $w_l \approx 0.01N$. While we previously introduced a compacity hypothesis, we still wish to be have a method which is robust in the event that its design hypothesis is not entirely met, a method that is able to process as many signals as possible even if they have a limited compacity. As a result, we will set the smoothing window size to $w_l = \lfloor 0.01N \rfloor$. Note that for higher signal compacity applications, a higher value of w_l can be set. This will improve the estimator performance, as shown earlier.

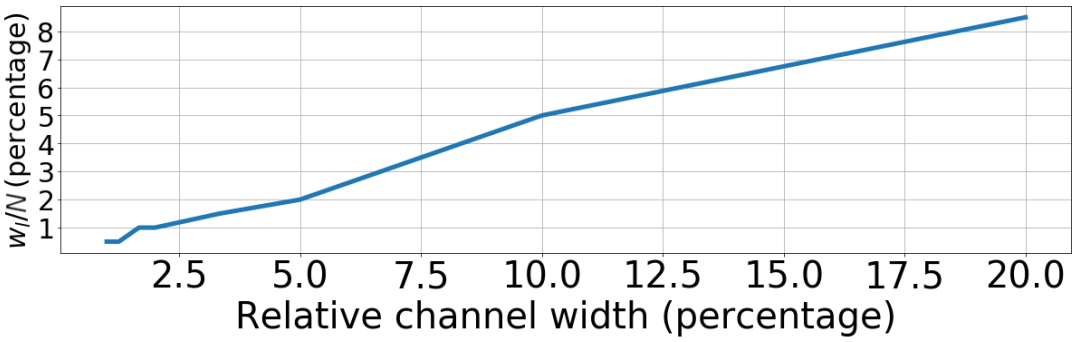


Figure 2.8: Maximum relative smoothing window size vs relative channel width, subject to estimation performance constraints.

Note that extremely low compacity signals (relative channel width under 1%) can only be processed with very small smoothing windows, jeopardizing the results of the method for $N = 1000$ samples; however, noise variance estimation can be improved using higher numbers of samples N, at the expense of a loss in time resolution.

2.2.4.4 Application to a telecom signal: the filtered QPSK

Finally, we want to test the proposed method with a more realistic SoI than the "constant energy" SoI previously used. For this purpose, we use the same simulation conditions as in Section 2.2.4.2, except that in each channel of the frequency support, the signal of interest X is a random character string modulated by a filtered QPSK with roll-off factor $\beta = 0.35$. X is subsequently normalized and multiplied by $\sqrt{E}$ in order to obtain the desired SNR. The simulation is run 50 times and the median value is returned.

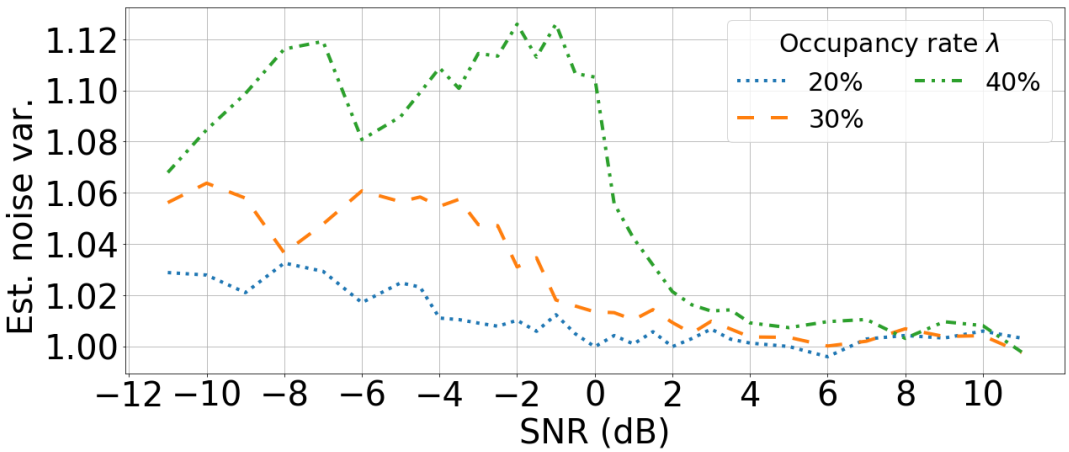


Figure 2.9: $\hat{\sigma}_w^2$ vs SNR for various occupancy ratios. The SoI X is a sum of filtered QPSKs.

Fig. 2.9 displays the results of the simulation for occupancy ratios ρ ranging from 20% to 40% and for SNRs between -11 dB and 11 dB. The first observation is that compared to the simulation in Section 2.2.4.2, the transitions between the different regimes are less clear. For lower occupancy ratios (under 40%), the estimator performs fairly well, although the intermediary regime seems larger than in the "constant energy" simulation. For the lowest occupancy

ratios (20% and under), the estimator is very robust to very low SNRs. For higher occupancy ratios (40% and above, which are not depicted), the "high SNR" regime, where the estimator has the highest success rates, is reached at higher SNRs than in the "constant energy" simulation. Yet, the relative error in the low SNR/intermediary regime for a 40% occupancy ratio remains at a reasonable level, around 10%. This suggests that the method struggles to perform correct clustering for high occupancy ratio, low SNR signals. This is most likely due to the structure of the SoI: the values of $|X|^2$ are less grouped together than in Section 2.2.4.2, making it harder for the proposed method to conduct clustering on the energy of samples.

2.2.5 Conclusion and Perspectives on Noise Variance Estimation

We proposed a novel application of the K-means algorithm, which solves the problem of blind noise variance estimation in an AWGN channel, under two hypotheses of signal sparsity and compacity. This method is robust, has a low complexity (each step has a linear or constant complexity) and requires a reasonable amount of samples.

While the AWGN channel is a rather simplistic model for terrestrial communications, it is useful to simulate background noise, on which our estimator focuses. In practice, more complex impairments, such as fading, shadowing or interference, may affect the signal of interest, but not the level of the background noise. As a consequence, our estimator is still relevant even in a richer, closer-to-reality model – as long as there is background noise in which the signal of interest lies.

Improvements in the method's performance could be obtained either through a finer tuning of its parameters or through slight changes in the structure of the method. For example, this could consist in choosing another smoothing window, in adding a feedback loop for the compacity level of the input signal to adaptively update w_l. The gains would either be in terms of number of required samples or in robustness (to absence of sparsity/compacity).

Here, we focused only on a one-dimensional signal sparsity, that is, in the frequency domain (though the method is also directly applicable to sparsity in the time domain). However, the method could be extended to take advantage of higher-dimensional sparsity: for example, exploit a two-dimensional sparsity, in both the time and frequency domains. Further, other types of clustering, like density-based clustering, could be explored instead of the centroid-based K-means.

2.3 Support Estimation

2.3.1 Introduction to Support Estimation

Now that we are able to blindly estimate the AWGN variance σ_w^2, we can turn to support estimation. Note that the outcome of the support estimator will be used later when the MRS is considered.

2.3.1.1 Previous works

The support estimation problem, framed in [181, 182], is related to the well-known problem of signal detection [139]. Several approaches to support estimation in a wideband signal involve successive "channel-by-channel" signal detections ([163, 183, 184, 185]). These are quick and convenient for dynamic spectrum access because communication can take place as soon as an empty channel is identified. However, for full support recovery, the procedure involves signal detection in all channels, which takes more time, as mentioned in Chapter 1. Furthermore, these solutions have granularity and slotting issues and can underperform if the channel width for detection is unadapted.

Cai and Wang [15] explored an energy-based method and introduced a criterion for non-zero signal values to be detected with a given probability. However a parameter of the method depends on the level of spectral occupation of the signal, which is not necessarily known.

2.3.1.2 Principle of our method and section outline

We now propose a novel method for signal support estimation of a sparse wideband signal corrupted by an AWGN with known variance (see our proposed method for blind noise variance estimation in Section 2.2). Our method consists in an enhanced sample-wise energy detection based on the Neyman-Pearson (NP) theory. We present our method in the context of uniform sampling: our findings can also be used for regular SS with legacy, Nyquist sampling.

Throughout this section, we will perform the following. First, in the context of the support estimation problem, we will derive closed-form expressions for several thresholds and quantities stemming from the NP theory. Second, we will confirm and discuss our findings through simulations. Third, we will apply our support estimation method to realistic telecommunication signals, compare the outcome with the theoretical results and draw conclusions on the efficiency of the method.

As we will see, this study shares some similitudes with that of Section 2.2, regarding the system model for instance. The updated system model for support estimation is presented in Section 2.3.2. Section 2.3.3 investigates the sample-wise energy threshold for support estimation and explores optimal signal values for correct support estimation. Section 2.3.4 proposes an enhancement, based on window smoothing, of our support estimation method. This enhancement is similar to the preprocessing step of our noise variance estimator. In Section 2.3.5, our theoretical findings are evaluated against empirical observations. Section 2.3.6 puts forward an application of our method to sparse telecommunication signals.

2.3.2 System Model

In this study, we keep the naming conventions and definitions of Section 2.2.2. We now introduce additional quantities and concepts pertaining to this study.

The frequency-domain support K of the signal X is the subset of $[\![0, N-1]\!]$ that contains all the indices j so that $X[j]$ is non-zero.

$$K = \{j \in \{0, 1, \ldots, N-1\}, X[j] \neq 0\}. \tag{2.8}$$

The purpose of the method is to recover the unknown signal support K from noisy signal vector Y.

Let $||X||_0 = \text{Card}(K)$ be the ℓ_0-pseudonorm of X. In other terms, this pseudonorm is the number of nonzero elements in X. The signal X is said to be sparse if $||X||_0 \ll N$.

In contrast to the system model of the noise variance estimator, the variance σ_w^2 of the AWGN is supposed to be known. For simplicity and without loss of generality, we also suppose that $\sigma_w^2 = 1$. Were it not the case, Y should be divided by $\sqrt{\sigma_w^2}$ to obtain a AWGN channel with normalized variance.

In Sections 2.3.3 to 2.3.5, the SoI X is supposed to be constant inside its support:

$$X[i] = \sqrt{E}e^{j\pi/4}, i \in K, \tag{2.9}$$

where E is the energy of each sample of X inside its support. This simplifying assumption is not necessary in any way. However, it allows each sample of Y to be modeled with simple, usual distributions that typically require many signal samples [139]. It also eases the discussion on the minimal level of energy of each sample to avoid signal misdetection, and makes it possible to discuss the link between the SNR and the signal occupancy ratio (or sparsity). Simulation results with more realistic SoIs will be presented in Section 2.3.6.

2.3.3 Sample-wise energy-based signal detection

2.3.3.1 Reminder on energy detection

As mentioned in Section 1.4.2.1, energy-based signal detection is an application of statistical hypothesis testing. The two hypotheses are the null hypothesis $\mathcal{H}_0$ (absence of signal) and the alternative hypothesis $\mathcal{H}_1$ (presence of signal). To make a decision on absence or presence of signal, a test statistic $T_{ED}(Y)$ is compared to a threshold η:

$$T_{ED}(Y) \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \eta. \tag{2.10}$$

Under NP theory, the test statistic $T_{ED}(Y)$ is proportional to the energy of Y [138].

Gains in detection accuracy are partly provided by increasing the number of values in the array to test. Hence, detection is usually conducted on a large number of samples, at the expense of a lower resolution in time or frequency. While this is not a serious problem if energy detection in a "large" bandwidth is considered, it is not suitable for signal support detection because of the loss in frequency resolution. For our purpose, we will perform single-sample signal detection.

2.3.3.2 Sample-wise detection

Our proposed method consists in a sample-wise energy-based signal detection, performed on each sample of the frequency support.

Let us consider the signal Y described in (2.1). If a frequency sample is in K, then it contains both SoI and noise. Otherwise, if the sample is not in K, it only contains noise:

$$\forall i, \begin{cases} \mathcal{H}_0(i \notin K) : |Y[i]|^2 = |W[i]|^2 \\ \mathcal{H}_1(i \in K) : |Y[i]|^2 = |X[i] + W[i]|^2 \end{cases} . \tag{2.11}$$

Since W is an AWGN and X is constant in its support with energy E in each sample (split evenly between the real and imaginary parts), the test statistic is as follows:

$$\forall i, \begin{cases} \mathcal{H}_0(i \notin K) : 2|Y[i]|^2 \sim \chi^2(2) \\ \mathcal{H}_1(i \in K) : 2|Y[i]|^2 \sim \chi^2(2, 2E) \end{cases} , \tag{2.12}$$

where $\chi^2(k)$ (resp. $\chi^2(k, \lambda)$) is the central (resp. non-central of non-centrality parameter λ) Chi-squared distribution with k dof. Note that these equations were already presented by Urkowitz [139], but for the purpose of energy detection: in [139], the Chi-square distributions resulted from an approximation on many samples, whereas here we consider only one complex sample.

We wish to find the threshold η that separates the noise outside of the support from the SoI-plus-noise mixture. Let $(m, n) \in \{0, 1\}^2$. Consider the decision $\mathcal{H}_m|\mathcal{H}_n$ ($\mathcal{H}_m$ is decided while $\mathcal{H}_n$ is true), taken on one sample. In binary decision, there are two successful outcomes ($\mathcal{H}_1|\mathcal{H}_1$ and $\mathcal{H}_0|\mathcal{H}_0$) and two unsuccessful outcomes: $\mathcal{H}_1|\mathcal{H}_0$ (false positive or type I error) and $\mathcal{H}_0|\mathcal{H}_1$ (false negative or type II error).

A common procedure when performing signal detection using the NP theory is to set the desired probability of false alarm P_{FA}, then to compute the threshold η using a closed-form expression, such as (1.58) of Chapter 1, derived from a system of equations. However, these equations rely on the assumption of a high number of samples, so that the Chi-square distributions under $\mathcal{H}_0$ and $\mathcal{H}_1$ can be approximated by Gaussian distributions using the central limit theorem [138]. On the contrary, our decision is taken on only two values (the real and imaginary parts of a given complex sample), rendering the aforementioned closed-form expression inadequate for our purpose. To handle this issue, we resort to the original Chi-squared expression of the test statistic.

2.3.3.3 Link between the threshold η and P_{FA}

Under $\mathcal{H}_0$, the threshold η is simply half[3] the (1-P_{FA})th quantile of the central Chi-squared distribution with two dof. This is a direct consequence of the definition of a quantile:

$$P(2|Y[i]|^2 < 2\eta) = 1 - P_{FA}. \tag{2.13}$$

The cumulative distribution function (cdf) $F(x; 2)$ of the $\chi^2(2)$ distribution is the following:

$$F(x; 2) = 1 - e^{-x/2}. \tag{2.14}$$

Combining (2.13) and (2.14) yields the expression of the threshold η as a function of P_{FA}:

$$\eta = -\log P_{FA}. \tag{2.15}$$

[3]Indeed, it is $2|Y[i]|^2$ that follows a central Chi-squared distribution with two dof, not $|Y[i]|^2$.

Through this framework, type I errors can be controlled and set to any given level. However, the fewer the type I errors, the higher the threshold η, and the more the type II errors. Type II errors are problematic when support estimation is performed as a preprocessing step for further operations on a signal. For this reason, we need to analyze what the energy of the SoI should be in order to control type II errors.

2.3.3.4 Mitigating Type II errors

Let $E_{\min}$ represent the energy that a (noiseless) sample of the SoI X should have, so that the outcome of the sample-wise detection on the noise-contaminated sample has a fixed, given probability of a type II error. Under $\mathcal{H}_1$, the problem is the following: find $E_{\min}$ so that

$$P(2|Y[i]|^2 < 2\eta) = 1 - P_D \text{ with } 2|Y[i]|^2 \sim \chi^2(2, 2E_{\min}), \tag{2.16}$$

where P_D is the probability of detection. Note that $P_M = 1 - P_D$ is the probability of misdetections, or type II errors.

From the cdf $F(x; k, \lambda)$ of the $\chi^2(k, \lambda)$ distribution and (2.16), we obtain a relationship[4] between P_D, η and $E_{\min}$:

$$P_D = 1 - e^{-E_{\min}} \sum_{j=0}^{+\infty} \frac{{E_{\min}}^j}{j!} Q\left(1 + j, \eta\right), \tag{2.17}$$

where $Q(a, x)$ is the regularized lower incomplete gamma function, defined as:

$$Q(a, x) = \frac{1}{\Gamma(a)} \int_0^x t^{a-1} e^{-t} dt \tag{2.18}$$

with $\Gamma(a)$ the gamma function.

Combining (2.15) and (2.17) yields a relationship between P_D, P_{FA} and $E_{\min}$:

$$P_D = 1 - e^{-E_{\min}} \sum_{j=0}^{+\infty} \frac{{E_{\min}}^j}{j!} Q(1 + j, -\log P_{FA}). \tag{2.19}$$

A numerical computation is then carried out in order to obtain $E_{\min}$. Receiver Operating Characteristic (ROC) curves for varying values of $E_{\min}$ are depicted in Fig. 2.11 of Section 2.3.5. As $E_{\min}$ increases, the ROC curve approaches the upper left-hand corner of the $P_D - P_{FA}$ plane. As one could have expected, a higher energy in each SoI sample results in better detection performance.

2.3.3.5 Link between P_D, P_{FA}, occupancy rate and SNR

From (2.19), setting P_D and P_{FA} also sets the value of $E_{\min}$, that is, the energy of one frequency sample to reach the fixed values of P_D and P_{FA}.

The energy in one sample $E_{\min}$ is proportional to the linear SNR by the following relation:

$$SNR = \frac{E_{signal}}{E_{noise}} = \frac{||X||_0 E_{\min}}{N\sigma_w^2}, \tag{2.20}$$

[4]How this relationship is obtained is detailed step-by-step in Appendix A.

where the noise variance is $\sigma_w^2 = 1$.

Let $\rho = \frac{||X||_0}{N}$ be the spectral occupancy ratio. (2.20) becomes

$$SNR = \rho E_{\min}. \tag{2.21}$$

As a consequence, using (2.17) and (2.21), it is possible to link SNR, P_D, P_{FA} and ρ.

These relations put forward the fact that relatively high SNRs (or equivalently, large values of $E_{\min}$) are needed for successful sample-wise energy detection. For example, with $P_D = 0.99$, $P_{FA} = 0.01$ and $\rho = 0.4$, a 7.4 dB SNR is required. Table 2.2 presents required SNRs for values of ρ.

ρ	0.1	0.2	0.3	0.4	0.5	0.7	0.9
SNR (dB)	1.4	4.4	6.1	7.4	8.4	9.8	10.9

Table 2.2: Required SNR (dB) and various occupancy ratios ρ for sample-wise detection with $P_D = 0.99$ and $P_{FA} = 0.01$.

In order to perform support estimation under lower SNRs, a moving average can be applied to $|Y|^2$.

2.3.4 Enhancing sample-wise energy detection with the moving average

2.3.4.1 The principle of the moving average

We now introduce the moving average as a way to perform support estimation under lower SNRs. Conceptually, it permits the smoothing of the noise-corresponding samples, which allows for lower thresholds η. In this regard, this step plays exactly the same role as the preprocessing step in the noise variance estimator of Section 2.2. However, this is at the expense of resolution, which is lowered when a moving average is performed. Note that a standard energy detection on an entire bandwidth corresponds to an extreme scenario of moving average applied to sample-wise detection.

Support estimation with a moving average differs from channel-wise signal detection because it is not subject to improper channel slotting, a situation where a signal on the edge between two channels does not contain enough energy in any of the two adjacent channels to be detected in either of the two channels.

Mathematically, moving average is performed by multiplying the signal by a window: we use the rectangular smoothing window, of length $w_l < N$ and of amplitude 1.

The moving average is applied to $|Y|^2$ as follows:

$$|Y^{ma}[i]|^2 = \sum_{k=\lfloor i-\frac{w_l}{2}\rfloor+1}^{\lfloor i+\frac{w_l}{2}\rfloor} |Y[k\%N]|^2, 0 \leq i < N. \tag{2.22}$$

The modulo function in the argument of the sum in (2.22) corresponds to the fact that after the last (resp. before the first) values of Y, the first (resp. the last) values of Y can be used again, since Y is periodic as the output of a discrete Fourier transform.

As a consequence of the application of a smoothing window, under $\mathcal{H}_0$, realizations of $2|Y^{ma}[i]|^2$ no longer follow Chi-square distributions with 2 dof, but with $2w_l$ dof. Subsequently, the new test statistic is the following:

$$\forall i, \begin{cases} \mathcal{H}_0(i \notin \mathbb{K}) : 2|Y^{ma}[i]|^2 \sim \chi^2(2w_l) \\ \mathcal{H}_1(i \in \mathbb{K}) : 2|Y^{ma}[i]|^2 \sim \chi^2(2w_l, w_l E) \end{cases} . \tag{2.23}$$

2.3.4.2 Link between the threshold η and P_{FA}

The cdf $F(x;k)$ of the $\chi^2(k)$ distribution is the following:

$$F(x;k) = Q(\frac{k}{2}, \frac{x}{2}). \tag{2.24}$$

Replacing Y by Y^{ma} in (2.13) and combining the latter with (2.24) yields:

$$\eta = Q^{-1}(w_l, 1 - P_{FA}), \tag{2.25}$$

where Q^{-1} is the reciprocal of the regularized lower incomplete gamma function. Note that setting w_l to 1 yields a reformulation of (2.15).

2.3.4.3 Mitigating Type II errors

Under $\mathcal{H}_1$, the problem is very similar to (2.16). We wish to find $E_{\min}$ so that

$$P(2|Y^{ma}[i]|^2 < 2\eta) = 1 - P_D \text{ with } 2|Y^{ma}[i]|^2 \sim \chi^2(2w_l, 2w_l E_{\min}). \tag{2.26}$$

The equivalent of (2.17) with the moving average is:

$$P_D = 1 - e^{-w_l E_{\min}} \sum_{j=0}^{+\infty} \frac{(w_l E_{\min})^j}{j!} Q\left(w_l + j, Q^{-1}(w_l, 1 - P_{FA})\right). \tag{2.27}$$

Again, this relation is solved for $E_{\min}$ with fixed P_D and P_{FA} through a numerical computation. ROC curves for varying values of w_l are depicted in Fig. 2.12 of Section 2.3.5. As $E_{\min}$ increases, the ROC curve approaches the upper left-hand corner of the $P_D - P_{FA}$ plane.

Unsurprisingly, the longer the smoothing window, the higher the performance of the detector. This is another illustration of a decision on presence or absence of energy taken on more samples and therefore more accurate. Note that higher values of both $E_{\min}$ and w_l can combine constructively to yield ROC curves even closer to the upper left-hand corner of the $P_D - P_{FA}$ plane.

2.3.4.4 Effect of the smoothing window length on required SNRs

Similarly to what happens in ED, increasing the number of samples improves the estimator's performance.

Figure 2.10 displays the tradeoff between window length and SNR, for different (P_D, P_{FA}) pairs and an occupancy ratio ρ of 40%. We can see that a small window size $w_l < 10$ samples already results in a substantial gain in SNR. For example, a 3 to 4 dB gain in SNR is obtained by using a window length of 3, compared to no windowing ($w_l = 1$).

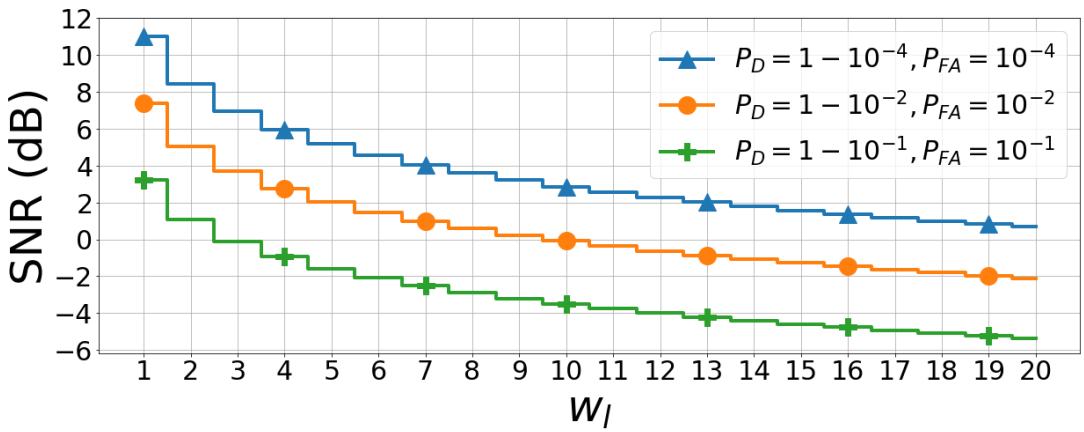


Figure 2.10: Required SNR (dB) versus window length w_l, for various (P_D, P_{FA}) pairs. The occupancy ratio is $\rho = 40\%$.

2.3.4.5 "Neither Hypothesis" samples: a side effect of windowing

When performing a moving average in support estimation, a side effect takes place. Indeed, at the edges of the intervals inside the SoI support, averaging causes samples of Y both inside and outside the support to be associated in (2.22). This results in samples of Y^{ma} for which neither $\mathcal{H}_0$ nor $\mathcal{H}_1$ is true. We will refer to these side-effect samples as *neither hypothesis* (NH) samples. By definition, a NH sample follows neither of the two Chi-squared distributions of (2.26). Consequently, deriving closed-form expressions such as (2.25) and (2.27) is less tractable for NH samples. Intuitively, a decision taken on a NH sample is less predictable and therefore closer to a random decision than a decision taken on a non-NH sample. Having too many NH samples is detrimental for support estimation because the binary decision on these samples is ultimately unreliable.

Actual performance of support estimation on NH samples depends on the energy in these NH samples and though equations could potentially be tractable in our simplistic model, we deem them to be too specific to provide useful insight for more realistic models. Instead, to quantify the magnitude of this side effect, we will consider the fraction ν of NH samples in a sampled signal Y:

$$\nu \triangleq \frac{\text{Card}\{\text{NH samples}\}}{N}. \tag{2.28}$$

We identified four factors that can make ν go up, and now present them using qualitative argumentation. For simplicity, every factor is discussed with all other parameters being equal. First, a lower compacity (characterized by smaller intervals in K) results in more intervals in K. In turn, there are more support edges and thus more samples at the support edges. Second, following the same reasoning, a higher occupancy ratio also increases the number of intervals in K and thus results in more samples at the support edges. Third, a higher window length increases the number of NH samples at each support edge. Fourth, a lower number of samples n (for a given acquisition time) reduces the frequency resolution: every component of the SoI is represented with fewer frequency samples. Consequently, each interval in K contains fewer elements, resulting in more NH samples.

Out of these four identified factors, the compacity and occupancy ratio are not under our control and cannot be changed, the number of samples can be amended but not without altering sampling rate and/or duration, and the window length is easier to modify.

We now bound ν. Let l be the average size of an interval in K and $c = l/N$ (c stands for compacity and is a basic measure of the compacity of SoI X). Let ι be the number of intervals in K. Because each interval yields up to $2w_l$ NH samples, we have

$$\nu \leq \frac{2\iota w_l}{N}. \tag{2.29}$$

Numerical example: for a number of samples $N = 1000$ samples, a window length $w_l = 10$ samples, an average interval length $l = 100$ samples (yielding a compacity $c = 0.1$) and a number of intervals $\iota = 3$ intervals (yielding an occupancy ratio $\rho = 0.3$), from (2.29), we have a fraction of NH samples $\nu \leq 0.06$.

Since on average $\iota l = ||X||_0$, we have $\rho = \iota l/N = nc$. Thus, (2.29) becomes

$$\nu \leq \frac{2\rho w_l}{Nc}. \tag{2.30}$$

Exact values of ν depend on the actual positions of the SoI components in the spectrum, and are lower than the upper-bound (2.29) if either an interval of K or a white space between two consecutive intervals of K are smaller than $2w_l$ samples.

Theoretically quantifying the degradation caused by a high fraction ν of NH samples can be tenuous. For now, we make a reasonable assumption that having NH samples shifts P_D and P_{FA} towards the center of the $P_D - P_{FA}$ plane, thus worsening the performance of the estimator. We also suppose that the higher the value of ν, the stronger the shift. In the following section, we will provide empirical evidence of this shift.

2.3.5 Simulations: support estimation

To perform support estimation, we perform sample-wise energy detection on every sample in the vector, with ($w_l > 1$) and without ($w_l = 1$) windowing.

2.3.5.1 Performance metric

Each sample of $|Y|^2$ (or $|Y^{ma}|^2$) is compared to a threshold η, computed using (2.25). The result is the estimated signal support, a N-sized boolean vector. To evaluate the performance of our estimator, we use p_d and p_{fa}. Since we are not considering random variable distributions but realizations, p_d and p_{fa} are not probabilities but proportions, defined as follows:

$$p_d = \frac{d}{||X||_0}, p_{fa} = \frac{fa}{N - ||X||_0}, \tag{2.31}$$

where d (resp. fa) is the number of samples for which $\mathcal{H}_1|\mathcal{H}_1$ (resp. $\mathcal{H}_1|\mathcal{H}_0$) is decided.

If $\hat{K}$ denotes the estimated support, another formulation of p_d is

$$p_d = \frac{\text{Card } K \cap \hat{K}}{\text{Card } K}. \tag{2.32}$$

2.3.5.2 Simulation results for different values of $E_{\min}$

First, we compare empirical performance of our estimator for various values of $E_{\min}$ without windowing ($w_l = 1$). Fig. 2.11 displays the ROC curves for varying values of $E_{\min}$ (in arbitrary units). The constant parameters for the simulation are $N = 2000$, $\rho = 40\%$, and $c = 10\%$. Dashed lines correspond to theory presented in Section 2.3.3 while solid lines display empirical results. Each simulation is averaged over 500 runs.

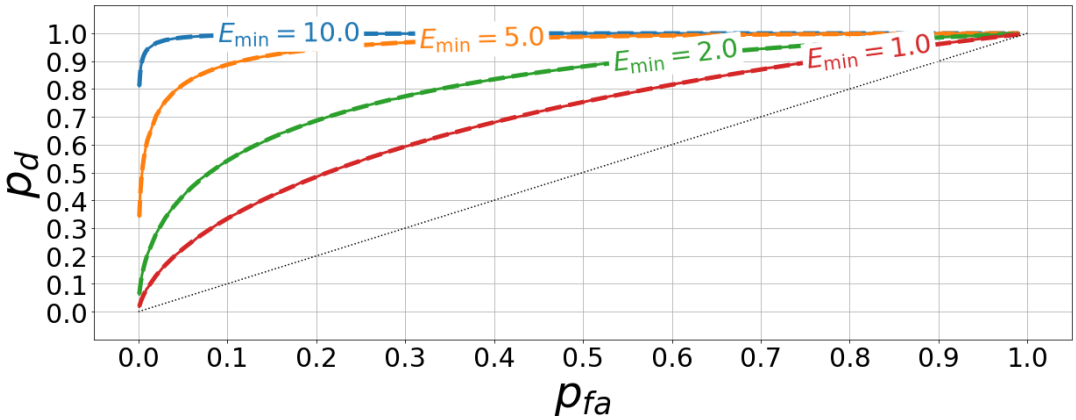


Figure 2.11: ROC curves of empirical sample-wise energy detection for varying values of $E_{\min}$.

The empirical results perfectly match the theory. In our model, the SNR and ρ are linked to $E_{\min}$ by (2.21), but have no other specific influence on estimator performance. Consequently, they will not be studied explicitely in the remainder of this section. An exception to this is the role played by ρ in the $w_l > 1$ case, where it affects the number of NH samples, as seen in (2.30). This topic will be covered in Section 2.3.5.4.

For $E_{\min} = 1.0$ (arbitrary unit), corresponding to a SNR of -4.0 dB in this setup, the performance of our support estimator is low. We now consider the effect of using the moving average to improve estimator performance.

2.3.5.3 Simulation results for varying window lengths

Fig. 2.12 displays the ROC curves for varying window lengths. The constant parameters for the simulation are $N = 5000$, $\rho = 40\%$, SNR $= -4.0$ dB and $c = 5\%$. The case $w_l = 1$ corresponds to non-averaged support estimation. Dashed lines correspond to theory presented in Section 2.3.4 while solid lines display empirical results.

We observe that for the general trend, the experimental results validate the theory. In both cases, estimation can be very good: for instance, a value of $P_{FA} = 0.1$ allows for a theoretical $P_D = .977$ and an empirical $p_d = 0.964$. Note that estimation performance does not depend on the SNR directly, but on $E_{\min}$. Eq. (2.21) allows for similar estimation performance for various values of SNR and ρ, as long as $E_{\min}$ is unchanged. As a consequence, estimation performance is the same for SNR $= -4.0$ dB and $\rho = 0.4$ than for SNR $= -10$ dB and $\rho = 0.1$. This opens the way to high estimation performance for low-SNR and very sparse signals.

For $w_l = \{10, 20\}$ samples, the empirical results diverge slightly from the theoretical ROC

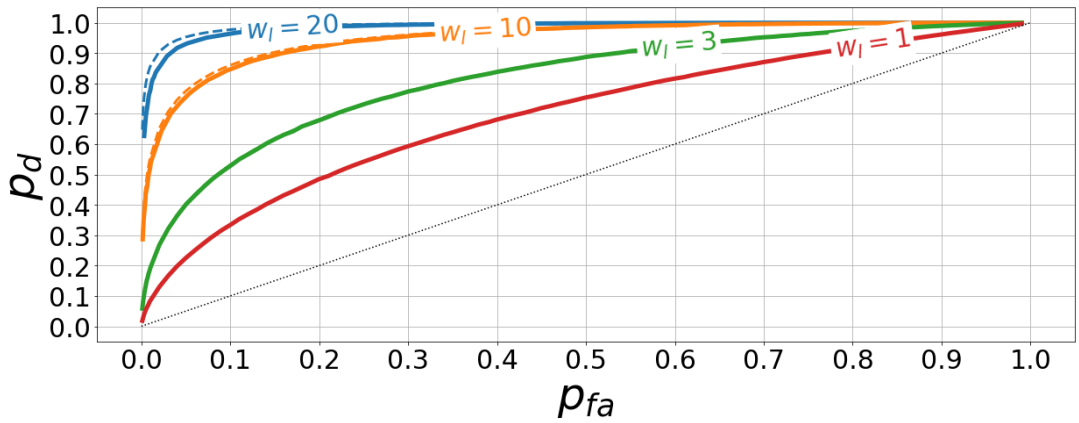


Figure 2.12: ROC curves of sample-wise energy detection with moving average and varying window lengths w_l.

curve: this is an illustration of the detrimental effect caused by NH samples. Fractions ν of NH samples are respectively 2.9% and 6.0% for $w_l = \{10, 20\}$ samples.

2.3.5.4 Degradation caused by NH samples

The fraction ν of NH samples is a basic indicator of the presence of discrepancies between theoretical and empirical results, and quantifying the performance degradation caused by a high value of ν is difficult when all parameters are taken into account at the same time. Nonetheless, we wish to document, even partially, the degradation caused by NH samples. Fig. 2.13 displays the ROC curves for varying values of ν. The constant parameters are $N = 2000$, $\rho = 40\%$, SNR $= -4.0$ dB and $w_l = 20$. The dashed lane represents the theoretical estimator performance (for which $\nu = 0$). To alter the fraction of NH samples, we changed the compacity c of the SoI. Indeed, the closed-form expressions of Section 2.3.4 do not involve c, which only comes into play when NH samples are concerned.

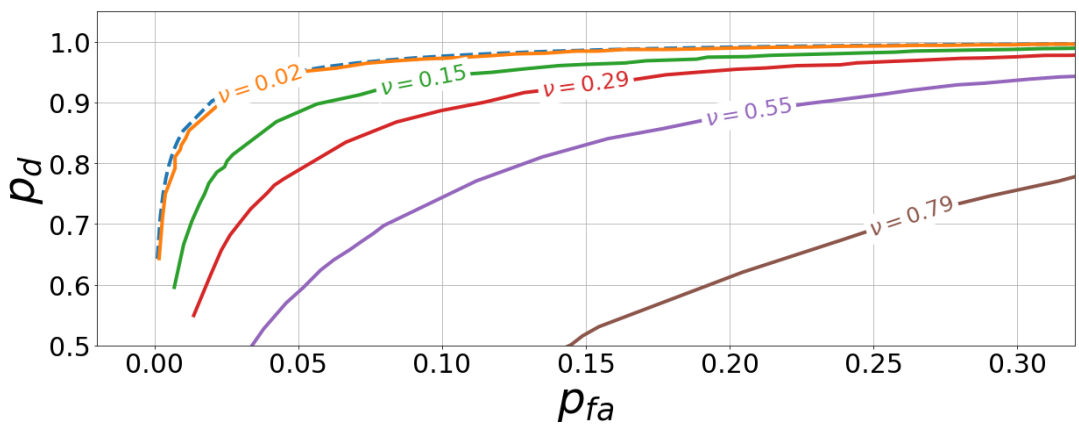


Figure 2.13: ROC curves of sample-wise energy detection with moving average for varying fractions of NH samples ν.

For this scenario, each increase of the fraction of NH samples by 0.1 results in an estimator performance degradation that is equivalent to a reduction of approximately $0.6 - 0.7$ dB of SoI

SNR. This figure varies with N, the SNR and w_l.

To limit the influence of the moving average side effect, we recommend having $\nu \leq 0.1$. Table 2.3 provides recommended window lengths for several values of N, compacity c and occupancy ratios ρ. These values are computed using (2.30). The (c, ρ) pairs represent different scenarios, from the easiest to process (high compacity, low occupancy ratio which corresponds to few users each transmitting a wide component of the SoI) to the hardest to process (low compacity, high occupancy ratio: many users transmit narrow signals to form the SoI at the receiver). As the transmission scenario becomes less favorable, the required w_l becomes smaller, entailing overall lower performance of the support estimator. In a realistic scenario, (c, ρ) may not be known beforehand: a possible way to proceed is to first compute a rough support estimate with conservative parameters (high N if possible, low w_l), then to use this first support approximation to evaluate c and ρ and to tune w_l and N more finely.

	$\rho = .1$		$\rho = .3$		$\rho = .6$	
N	$c = .01$	$.05$	$c = .01$	$.05$	$c = .01$	$.05$
1000	5	25	1	8	1	8
2000	10	50	3	16	2	10
5000	25	125	8	41	5	25

Table 2.3: Recommended window lengths for a fraction of NH samples $\nu \leq .1$.

The performance of our estimator mostly matches the theory, and discrepancies have been investigated. However, all of our findings so far involve a simplistic signal model, where the SoI is constant in its support. As a consequence, we now study the performance of our estimator on more realistic telecommunications signals.

2.3.6 Application to a telecommunications signal: the filtered QPSK

2.3.6.1 A new signal model

For this round of simulations, the SoI X no longer follows (2.9). Instead, X is the sum of QPSK-modulated random messages. The frequency spectrum is slotted into 20 channels and each of these messages is in a different channel. Each QPSK goes through a root-raised cosine filter with roll-off factor $\beta = 0.35$.

Strictly speaking, X is never non-zero because of spectrum leakage. For this reason, we slightly adapt our definition of the support K, which now consists of the indices of X corresponding to the highest-energy samples of X. The threshold for the cumulative sum of highest-energy samples is $0.999||X||^2$, meaning that indices of low-energy samples collectively making up for less than 0.1% of the energy of X are discarded. For normalization purposes, we then multiply X by $\sqrt{E_{\min} \frac{\text{Card } K}{||X||^2}}$ so that on an average made on every sample in K, the energy in a sample in K is $E_{\min}$.

Because the SoI follows a different model, the closed-form expressions put forward in Sections 2.3.3 and 2.3.4 are no longer valid. Nonetheless, to validate the estimator performance

in practice, we will compare the results on QPSK-based SoIs to the theoretical ROC curves obtained through (2.27) (later referred to as the theory). This comparison is relevant because the model for SoI presented in (2.9) corresponds to a constant signal with the same sample-wise energy than the average sample-wise energy of the QPSK-based SoI.

To simplify the analysis and without loss of generality, in our experiment, each QPSK-based message (also referred to as a SoI component) contains the same energy. Roughly speaking, samples in the middle (resp. on the edges) of a SoI component have a higher (resp. lower) energy than the average energy $E_{\min}$ of a sample. Consequently, no matter the estimator setup, detection of the samples in the middle (resp. on the edges) of a component should in general be easier (resp. harder) than that of samples with average energy $E_{\min}$.

2.3.6.2 Support estimation performance: exact SoI support is hard to recover...

We have seen in Section 2.3.5 that the performance of the detector, represented by ROC curves in Figs. 2.11 and 2.12, is affected by variations in w_l and $E_{\min}$ in a similar fashion, meaning that a lower value of $E_{\min}$ can be compensated by a higher value of w_l and vice-versa. The same statement holds with another SoI model. For this reason, we will consider the window length w_l as the sole varying parameter. We chose to make w_l vary because in practice it is a parameter easier to tune than $E_{\min}$.

Fig. 2.14 displays the ROC curves for varying window lengths. The constant parameters for the simulation are $N = 2000$, $\rho = 40\%$, SNR $= -4.0$ dB and $c = 5\%$. Dashed lines correspond to constant SoI-based theory presented in Section 2.3.4 while solid lines display empirical results with QPSK-based signals. Each simulation is averaged over 100 runs.

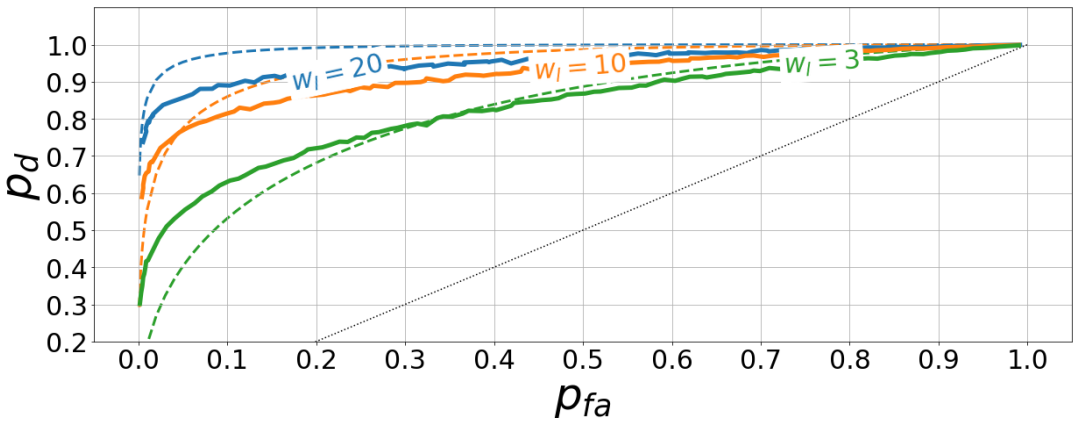


Figure 2.14: ROC curves of our support estimator with varying window lengths w_l, applied to (i) QPSK-based SoIs (solid lines); (ii) SoIs based on the constant energy model (dashed lines) .

For lower detection performance ($w_l = 3$ samples), the estimator outperforms the theory for a wide range of values of p_{fa}. We attribute this to the shape of the QPSK. While the estimator inconsistently decides $\mathcal{H}_1|\mathcal{H}_1$ in the constant SoI model due to high noise levels, the samples in the middle of the QPSK component are typically larger than $E_{\min}$ and stand out from the noise. As a result, the estimator decides $\mathcal{H}_1$ more consistently for these central samples. Meanwhile, the samples on the edge of the QPSK are typically under the noise floor: while the estimator

decides $\mathcal{H}_1$ even less for them than for constant SoI samples, it does not make a huge difference on overall estimation performance.

On the contrary, for higher detection performance ($w_l = 20$ samples), the estimator underperforms the theory for all values of p_{fa}. Again, this can be explained from the shape of the QPSK. When p_d is high, the comparative advantage of central QPSK samples over constant SoI samples is no longer meaningful; however, samples on the edge of the QPSK component, being lower than $E_{\min}$, are more often under the estimator threshold than in the constant SoI theory, and the estimator decides $\mathcal{H}_1$ less frequently on them, even as p_{fa} goes up.

Finally, the case $w_l = 10$ samples corresponds to the border between the two aforementioned regimes. Interestingly, the shape of the QPSK ultimately tends to reduce the theoretical gap between high and low estimator performances.

Overall, the performance of our estimator on detecting the precise support in which the SoI is located is below what we could expect from the theory, at least for high values of p_d. Fortunately, not all support indices are created equal: some SoI samples carry more imformation than others.

2.3.6.3 ... But we can recover most of the energy of the SoI

In filtered QPSK, as in many modulations, the information content is greater in the higher energy region, located at the center of the frequency-domain representation of the filtered QPSK, than in the lower energy region, located at the edges of the QPSK. Applications whose aim is to ultimately recover the signal information can take advantage of this fact and focus on estimating the location in the frequency spectrum of the highest energy samples. This observation can even be useful to a user that wishes to acquire knowledge about its radio surroundings without recovering the received signal.

In this regard, considering the SoI energy in the estimated support can be insightful. Since our estimator decides $\mathcal{H}_1$ more frequently for high-energy SoI samples than for average- and low-energy samples, we can expect the SoI energy in the estimated support to be higher for the QPSK-based SoI than for the constant SoI of (2.9).

Let $\hat{X}$ be the SoI X multiplied by the indicator function of the estimated support $\hat{K}$. $\hat{X}$ can be described as follows:

$$\forall 0 \leq i < N, \hat{X}[i] = \begin{cases} X[i] \text{ if } i \in \hat{K} \\ 0 \text{ otherwise} \end{cases} . \tag{2.33}$$

In other words, $\hat{X}$ is the SoI X over the estimated support $\hat{K}$. It is reminded that $\hat{X}$ is not obtainable in practice because the received signal is $Y = X + W$.

Let $e_d = ||\hat{X}||^2/||X||^2$ be the fraction of SoI energy in the estimated support $\hat{K}$ over the total SoI energy (we neglect the QPSK spectral leakage phenomenon mentioned earlier). In a way, e_d resembles p_d of (2.32), except p_d considers the cardinalities of K and $\hat{K} \cap K$ while e_d focuses on the energy of the samples in K and $\hat{K} \cap K$. Note that for the constant-energy SoI model (2.9), since every sample of X in K has the same energy, we have $p_d = e_d$. Finally, by definition of X, samples for which $\mathcal{H}_1|\mathcal{H}_0$ is decided have no energy in X.

Fig. 2.15 displays the fraction of detected energy e_d versus p_{fa} for varying window lengths. The constant parameters for the simulation are $N = 2000$, $\rho = 40\%$, SNR $= -4.0$ dB and $c = 5\%$. Dashed lines correspond to the constant energy SoI-based theory of (2.27) presented in Section 2.3.4 while solid lines display simulation results with QPSK-based signals. Each simulation is averaged over 100 runs.

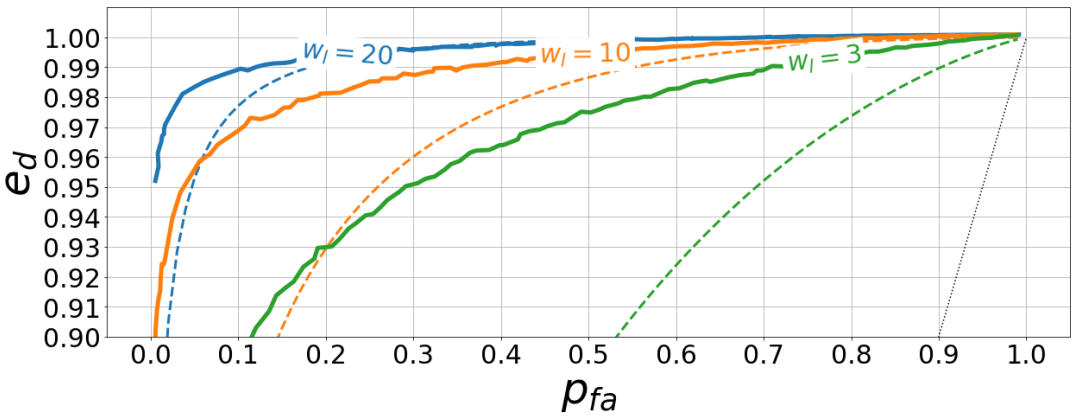


Figure 2.15: Fraction e_d of energy present in the support estimate $\hat{K}$ vs proportion of false alarms p_{fa} with varying window lengths w_l, applied to (i) QPSK-based SoIs (solid lines); (ii) constant energy SoIs (dashed lines).

As expected, for the QPSK-based SoI, $e_d \geq p_d$ for any given values of p_{fa} and w_l. A remarkable fact is that for every value of w_l, the solid curve is above the dashed curve: this means the estimator outperforms the theory, based on empirical results of e_d. Our estimator is great for deciding $\mathcal{H}_1$ for support elements that contribute significantly to the energy - and the information - of the SoI. For example, with a window length $w_l = 20$ samples and a proportion of false alarms $p_{fa} = 0.1$, the support estimated by our method contains 99.0% of the energy of the SoI X.

2.3.7 Conclusion on Support Estimation

In this section, we first presented a method for signal support estimation for noisy signals, based on sample-wise energy detection. Afterwards, we put forward an improvement to the method, consisting in a window smoothing preprocessing step. In both cases, we derived closed-form expressions for support estimation performance, for a simple (constant) model of signal of interest. Finally, we quantified the performance of our estimator on two types of signal of interest: a basic theoretical signal and a more realistic, QPSK-based signal.

Our method has a high performance for both types of signal and allows to estimate the frequency-spectrum location of an overwhelming proportion of the energy of a signal of interest. While being fast and simple, our estimator works well even in SNRs as low as -10 dB, without prior knowledge on the signal of interest. For these reasons, our estimator is particularly suitable as a preprocessing step for recovery of signals acquired through CS schemes such as the MRS.

We identified two directions in which our estimator can be improved upon. First, the relevance of adding a small-scale post-thresholding step should be studied as it can marginally

improve estimation performance. Such a step could for instance remove a single outlier in a large white space area from the support estimate, or conversely fill one- or two-sample holes in the support estimate. Second, we know that using the estimator on an unknown signal with cautiously picked parameters yields some information about the signal of interest. Applying the estimator to the received signal a second time, with finer parameters (*e.g.* the window length) computed from the first support estimation, should yield improvements in estimation performance that ought to be observed and quantified.

2.4 Conclusion

In this chapter, we have introduced a novel K-means-based technique to perform blind noise variance estimation on a sparse wideband signal in an AWGN channel. Afterwards, we put forward a new solution for frequency-domain signal support estimation. The output of the first method is used as an input for the second method, so both procedures are well interconnected.

Thanks to these two methods, we can extract the support of the signal of interest from uniform, sub-Nyquist signal samples, provided the signal of interest respects somewhat lax constraints of sparsity and compacity. In Chapter 4, they will be applied to signals sampled with the MRS, in a bid to recover the frequency-domain support of the Nyquist rate signal of interest, in a low-complexity fashion (*i.e.* without solving a CS-related USLE). In turn, this will allow us to separate samples corresponding to signal of interest from white noise samples. These "samples of interest" can then be stored for later customer-issued queries, as part of the envisioned use case for our sampling infrastructure.

In order to successfully recover the signal support from MRS samples, there are some necessary conditions on the parameters of the MRS scheme. Amongst these, the choice of the sampling rate of each MRS branch seems particularly important. The following chapter of this manuscript addresses this issue.

Chapter 3

Configuring the MRS settings

3.1 Introduction

A central element to our proposed sampling infrastructure is the chosen Compressed Sensing (CS) sampling scheme. We believe the Multi-Rate Sampling (MRS) to be adapted to our scenario of a low-cost, sub-Nyquist sampling infrastructure. Indeed, its hardware requirements are relaxed in the asynchronous flavor, and moderate in the synchronous version, rendering deployment with largely available hardware components possible in the near future.

The principal characteristics of any MRS implementation are the number of branches and the sampling rates in each branch. They directly affect the construction of the sampling matrix A of the CS Underdetermined System of Linear Equations (USLE) $y = Ax$ (1.15), where y is the measurement vector of size M, x is the original signal of size N and A is the sampling matrix of size $M \times N$. Consequently, correctly tuning these parameters is of utmost importance. It is to note that the authors of [20] do not specify how to choose the sampling rates, despite their influence on the MRS performance.

As we recall from Chapter 1, CS relies on two principles, according to the authors of [186]: *sparsity* and *incoherence*. Sparsity is the capacity of the signal of interest to be expressed in a very small number of coefficients in some basis Φ: this means that the signal of interest possesses much fewer degrees of freedom than its number of coefficients in some representation. Incoherence pertains to the method of sampling: incoherent sampling stipulates that contrary to the signal of interest, the waveforms used for sampling have a dense (non-sparse) representation in Φ. In the MRS, an improper choice of sampling rates can negatively impact the aforementioned incoherence of the sampling system.

This idea is somewhat developed in the coprime sampling literature. In [187], a pair of sampling periods is chosen to be coprime integers multiples of a common δT. The authors subsequently derive very interesting results for different applications, though none pertains directly to the reconstruction of sub-Nyquist-sampled signals. They also do not explore situations with three samplers or more.

In this chapter, we aim to bring the two approaches to this sampling paradigm, namely MRS and coprime sampling, closer together, as we believe coprime sampling to be a decisive tool towards higher MRS performance. We propose the sampling matrix rank as an indicator of the

recoverability of the original signal, superseding (for the MRS) the number of measurements M indicator, common in CS [55]. We also derive an upper-bound for the measurement matrix rank relying on the chosen sampling rates and the number of branches L, and present a condition on the choice of sampling rates to reach this upper-bound. Although the matrix rank is an interesting indicator, we observe that it is not perfect either, and that recovery performance depends not only on the sampling rates but also on the number of branches. Subsequently, we introduce graphical insights into sizing an MRS based on signal recovery quality requirements. The work presented in this chapter has been the subject of a peer-reviewed publication in an international conference [5].

Linking the MRS sampling rates to the sampling matrix rank is based on theoretical analysis, while relating the sampling rates (or rank) and number of branches to recovery performance is more experimental. We will first explore the mathematical relation between sampling rates and the sampling matrix rank, then turn to the examination of simulation results with varying parameters to draw insights regarding ways to adequately size an MRS.

3.2 Link between Sampling Rates and the Measurement Matrix Rank

3.2.1 Recovery guarantee properties do not characterize the typical behavior of a sampling setup

Previous work on sampling using random and deterministic matrices [38] put forward several properties (such as the *Restriction Isometry Property*, RIP, the *spark*, and the *coherence*) that guarantee the reconstruction[1] of x under certain conditions, usually on the level of sparsity k (number of nonzero values) of $X = DFT(x)$. Unfortunately, these properties require a high level of incoherence in the measurement matrix A. The sampling matrix of the MRS exhibits too much structure and correlations between columns for these properties to be satisfied for significant levels of sparsity. This is partly because properties have to guarantee recovery in the *worst-case scenarios*, which may happen with low or zero probability, in constrast to the *typical behavior* of a sampling system, for which a user might care more.

As we mentioned in Section 1.2.5 of Chapter 1, while these properties provide sufficient conditions for recovery, these conditions are by no means necessary, and USLEs not satisfying the aforementioned properties can be solved nonetheless. In particular, there is a documented gap between what the properties guarantee and empirical recovery performance [56, 55].

Let us consider "success" of a recovery procedure as an outcome with a low Mean Square Error (MSE) between original signal X and recovered signal estimate $\hat{X}$, and a "failure" as a different outcome. Let $\delta = M/N$ be the undersampling ratio and $\rho = k/M$ be the occupancy ratio. If we consider the recovery success rate on a $\rho - \delta$ plane, there is evidence that there are well-defined success and failure phases for recovery (see Fig. 1.4 of Chapter 1). Furthermore, as the size of the problem N increases, the transition between the success phase and failure

[1] See Section 1.2.3 of Chapter 1 for a more extensive review of these properties.

phase is increasingly sharp. This means that a small variation of k, all other parameters being equal, can cause a drop from a 100% success rate to a 0% success rate. While these phases and transitions are documented for Gaussian sampling matrices, we believe that the same behavior can be observed with other sampling systems, such as the MRS. In Section 3.3, we will use empirical evidence to back this assumption.

3.2.2 Motivation for using the matrix rank as an indicator of the quality of recovery

For a variety of sampling schemes, including random-matrix-based schemes [38], the quality of recovery increases with the number of measurements M. However, for the MRS, a higher number of measurements M does not necessarily entail a better recovery if the sampling rates are not chosen properly.

This is why we propose the measurement matrix rank as a rough indicator of recoverability of the MRS-based system. By definition, the rank of a matrix A (written $\operatorname{rank} A$) is the dimension of the subspace generated by the columns of A. A high rank is a necessary condition for signal recovery. Indeed, when $\operatorname{rank} A$ is low, the columns of A generate a low-dimensional subspace, and the projection of X onto the subspace generated by the columns of A results in an irretrievable loss of information. Note that a high rank is not sufficient for signal recovery: if any two columns of A are linearly independent yet differ by a small angle, the problem may be ill-conditioned and recovery may be subject to numerical instabilities. Still, we suppose that the higher the rank, the better the recovery, and we will seek to maximize $\operatorname{rank} A$.

3.2.3 On the information acquisition process underlying the MRS

Let us consider the information acquisition process that underlies sampling in the temporal domain, that is, multiplication of the received signal with a Dirac comb function.

Each Dirac impulse (sampling instant) multiplied with the received signal corresponds to a projection of the signal onto an element of some basis $\mathcal{B}$ of a subspace $\mathcal{V}$ of the signal space. A Dirac impulse not exactly coinciding with any other Dirac impulse in the temporal domain increases the cardinality of $\mathcal{B}$ (and thus the dimensionality of the subspace $\mathcal{V}$, noted $\dim \mathcal{V}$) by 1. However, if two given Dirac impulses are superimposed (meaning they are exactly coinciding in the same place in the temporal domain), they project the signal onto the same element of the basis $\mathcal{B}$: as a result, $\dim \mathcal{V}$ does not increase. Because of this, $\dim \mathcal{V}$ is the number of non-overlapping Dirac impulses.

Now, the measurement matrix A is precisely a matrix that represents a homomorphism between the signal space and the subspace $\mathcal{V}$. By definition of the rank, $\operatorname{rank} A = \dim \mathcal{V}$. Therefore, $\operatorname{rank} A$ is the number of unique sampling instants.

This identity between $\operatorname{rank} A$ and the unique sampling instants lays the ground for the following mathematical results.

3.2.4 Sampling rates and matrix rank

Notation: while working on the MRS, we rely on the description of the MRS made in Chapter 1, Section 1.3.2.2.

We start with a simple result on the rank of the folding matrix F_i.

Lemma 3.1. *Let F_i be a $M_i \times N$ folding matrix of an L-branch MRS system, with M_i being the number of samples collected at branch i. Then*

$$\operatorname{rank} F_i = M_i. \tag{3.1}$$

Proof. Suppose N is even. Let $\{C_i\}_{1 \leq i \leq N}$ denote the set of columns of F_i. By construction, if $M_i \leq \frac{N}{2}$, we have $[\, C_1 \cdots C_{M_i} \,] = I_{M_i}$ (where I_n is the $n \times n$ identity matrix). If $M_i > \frac{N}{2}$, we have $[\, C_1 \cdots C_{\frac{N}{2}} \; C_{N-(M_i-\frac{N}{2}-1)} \cdots C_N \,] = I_{M_i}$. I_{M_i} is trivially of rank M_i. Reasoning for odd values of N is similar by replacing $\frac{N}{2}$ by $\frac{N-1}{2}$. □

This means that each folding matrix F_i folds the signal space (of dimension N) onto a subspace of dimension M_i, the number of samples collected at branch i.

Now, in the following theorem, we will bound $\operatorname{rank} A$.

Theorem 3.2 (Matrix rank upper-bound)**.** *Let A be the $M \times N$ sampling matrix of an L-branch MRS system, obtained as shown in Eq.* (1.39) *in Chapter 1. Then*

$$\operatorname{rank} A \leq M - (L-1). \tag{3.2}$$

Proof. By induction on L. Let $P(L)$ be the statement: $\operatorname{rank} A \leq \sum_{i=1}^{L} M_i - (L-1)$.

Base case: If $L = 1$, we have (Lemma 3.1) $\operatorname{rank} A = \operatorname{rank} F_1 = M_1 \leq \sum_{i=1}^{L} M_i - (L-1)$. So $P(1)$ is true.

Inductive step: Suppose $P(L)$. Let $A_k = \begin{bmatrix} F_1 \\ \vdots \\ F_k \end{bmatrix}$ for $k \geq 1$. We have (Lemma 3.1) $\operatorname{rank} A_{L+1} \leq \operatorname{rank} A_L + \operatorname{rank} F_{L+1} \leq \mathcal{Q}$ where $\mathcal{Q} = \sum_{i=1}^{L+1} M_i - (L-1)$ (the latter inequality is obtained through the induction hypothesis). However, there is a linear combination between the rows of A_{L+1}. Indeed, the sum of first M_1 rows of A_{L+1} is proportional (with ratio M_1/M_{L+1}) to the sum of its last M_{L+1} rows. This linear dependency prevents the quantity $\mathcal{Q}$ from being reached by $\operatorname{rank} A_{L+1}$: therefore, we have $\operatorname{rank} A_{L+1} \leq \mathcal{Q} - 1$. Hence, $P(L) \Rightarrow P(L+1)$.

□

This means that under our definition of MRS, the measurement matrix A can never be of full rank (except for the trivial $L = 1$ case). However, it can be only $L - 1$ units away from being full rank. If $L \ll M$ (a typical scenario), the rank deficiency is low and A can be almost full rank.

To pursue on the reasoning previously developed, the common divisor(s) of a pair (M_i, M_j) correspond to simultaneous Dirac impulses in branch i and j, as per Eq. (1.36). Whether or not the set of M_i is coprime, they all have 1 as a common divisor, so the first sampling instant is common to every sampling branch: every branch's first symbol is redundant but one. This explains the $(L-1)$ factor in (3.2).

We now present the equality condition of (3.2).

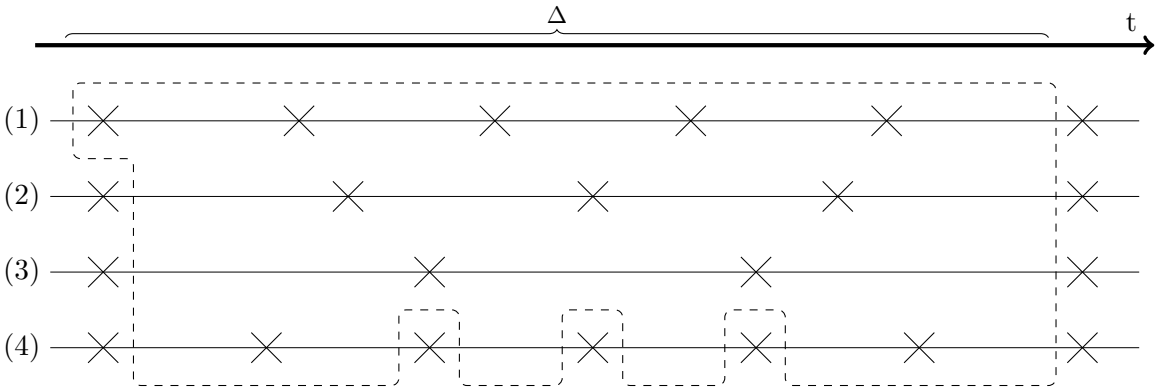


Figure 3.1: Multi-Rate sampling grid. This grid displays the Dirac impulses (sampling instants) for $L = 4$ branches with $M_i = 5$, 4, 3 and 6 respectively. During the duration Δ of the time window, only the samples comprised within the dashed area provide new information. Samples outside of the dashed area are redundant.

Theorem 3.3 (Equality condition)**.**

$$\operatorname{rank} A = M - (L-1) \Leftrightarrow \{M_i\}_{1 \leq i \leq L} \textit{ is pairwise coprime.} \tag{3.3}$$

Proof. Necessary part of the equivalence ($\Rightarrow$): By contraposition, we will prove that non-coprime values of M_i imply absence of equality. Suppose, without loss of generality, that $L = 2$ and that M_1 and M_2 are not coprime. Let D be the set of common divisors of (M_1, M_2). Let R_k^l denote the k-th row of folding matrix F_l. For every element $d \in D$, there is a linear combination between the rows of F_1 and the rows of F_2 (which is found again in A, the concatenation of F_1 and F_2): $\frac{1}{M_1}\sum_{i=1}^{M_1/d} R_{di}^1 = \frac{1}{M_2}\sum_{i=1}^{M_2/d} R_{di}^2$. This reduces the maximal value reachable by $\operatorname{rank} A$: the inequality of Theorem 3.2 becomes strict. (Note that the case $d = 1$ corresponds to the linear dependency already exhibited in the proof of Theorem 3.2 and is the cause of the inevitable small rank deficiency of A.)

Sufficient part of the equivalence ($\Leftarrow$): Having a pairwise coprime set of M_i allows to minimize the number of common divisors, leading to the maximization of the number of unique sampling instants and of $\operatorname{rank} A$. □

3.2.5 Illustration of the link between sampling rates and sampling matrix rank

Fig. 3.1 presents an example of an MRS grid with various values of M_i. The ticks within the dashed area correspond to the Dirac impulses which project the signal onto a new element of subspace $\mathcal{V}$ and provide new information about the signal. If we consider an MRS system with only the first three branches, all M_i are pairwise coprime, and every sample (except the first sample of branch 2 and 3) carries new information: $\operatorname{rank} A$ is maximal and Theorem 3.3 is verified. If we add the fourth branch (with $M_4 = 6$), then the pairs (M_2, M_4) and (M_3, M_4) are both non-coprime: this leads to redundancy and to a reduction of $\operatorname{rank} A$.

3.3 Empirical relations between MRS settings and recovery performance

3.3.1 Introduction

3.3.1.1 Scope of the simulations

The previous mathematical discussion pertains to the link between sampling rates and the matrix rank. We now focus on the parameters of the MRS.

The main parameters of the MRS are the number of branches L and the sampling rates $M_i, 1 \leq i \leq L$. In this section, we will provide empirical evidence that it is relevant to use the sampling matrix rank as an indicator of the recoverability of the original signal, rather than the number of measurements $M = \sum_i M_i$, common in the CS literature [55]. We will also quantitatively link the sampling matrix rank to the recovery performance of a MRS scheme.

Alone, the matrix rank indicator does not perfectly encompass the situation. In particular, for a given matrix rank, it appears that a higher number of branches L entails better recovery performance. We have not seen any mention of this phenomenon elsewhere in the literature. Though the causes of this phenomenon are unknown for the time being, our aim is to quantify this effect and use it to provide insights on the choice of the number of branches.

We will run our simulations on two kinds of received signals:

- k-sparse[2] noiseless signals, with $k < N$ non-zeros drawn according to a Gaussian distribution;
- Filtered QPSK-based noisy signals, where information-carrying QPSK-modulated spectral components are corrupted by AWGN.

The first type of signals enables us to benchmark MRS performance and compare it to other CS sampling schemes, while the second type shows the impact of the MRS settings in a more realistic situation.

For both kinds of signals, we will first focus on the rank as a relevant indicator, and then consider the impact of the number of branches L on signal recovery.

3.3.1.2 On the choice of metrics

Since our work is related to the sampling and recovery of a signal, the metric that first comes to mind is the Mean Square Error (MSE) between the actual received signal sampled at the Nyquist rate and the subsampled-then-recovered samples. When dealing with k-sparse noiseless signals, we use this metric to ascertain recovery success.

The MSE and its variants have the advantage of not depending on any external factor, such as choice of modulation or carrier impairments. However, it provides results that are rather difficult to interpret for a user whose motivation is to successfully transmit a message. In this regard, the Bit Error Rate (BER) is more explicit and therefore more relevant, despite being

[2] Signal sparsity is in the frequency domain.

dependent on the choice of the modulation scheme. We use the BER metric to present and discuss the results pertaining to the filtered QPSK-based noisy signals.

3.3.2 Simulation Results: Recovering k-sparse noiseless signals

3.3.2.1 Comparing the impact of sampling matrix rank and number of measurements on recovery

Simulation Setup A k-sparse signal X is generated in the following manner. A $N = 1024$-sized vector is initialized with zeros; then $1 \leq k < N$ of its elements are chosen randomly and each is set to a realization of a random variable following the $\mathcal{N}(0, 1)$ distribution. Signal X then undergoes an Inverse Fast Fourier Transform and is sampled by an MRS system with $L = 5$ branches. The sampling rates M_i are chosen at random so that $30 \leq M_i < 250$. After the sampling rates M_i are chosen, we construct the sampling matrix A and compute its rank. Based on whether or not the $\{M_i\}_i$ form a coprime set, the MRS system is either categorized as a quasi-full-rank (QFR) MRS or a deficient-rank (DR) MRS. Let us define the rank deficiency as $d = \operatorname{rank} A/M$ (where $M = \sum_i M_i$). For greater result readability, we chose to discard DR-MRSs for which $d > 0.90$. Indeed, such high values of d mean that even if the $\{M_i\}_i$ are not coprime, the matrix rank is very close to the upper-bound in (3.2). Finally, the reduction procedure proposed in [20] is applied, and an estimate $\hat{X}$ of the original signal X is computed using the CVXPY convex optimization solver. The relative RMSE (Root MSE) $||\hat{X}-X||_2/||X||_2$ is computed. The signal recovery is considered successful if the relative RMSE is under 1% and unsuccessful otherwise.

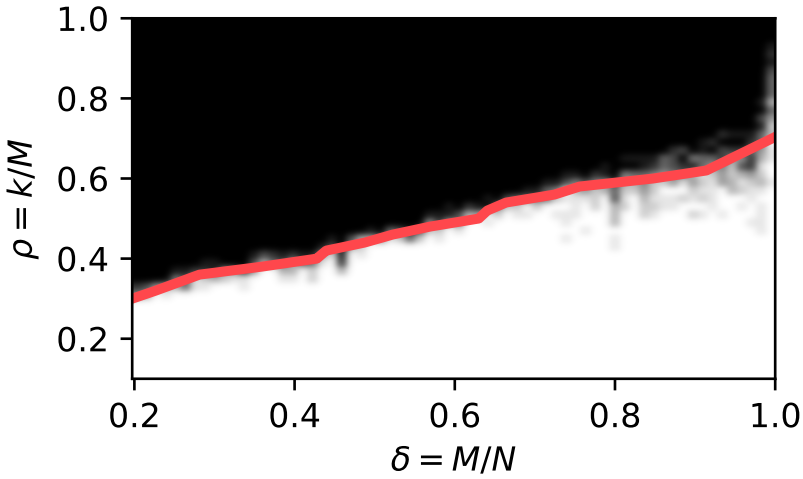
In total, 388 MRS systems (66 QFR, 322 DR) are tested for ratios $\rho = k/M$ ranging from 0.01 to 1. The simulation is averaged over 10 runs.

Phases and transitions Fig. 3.2 displays the success rate of recovery on the $\rho - \delta$ plane (the lighter the hue, the higher the success rate), for both QFR-MRS (panel (a)) and DR-MRS (panel (b)). The solid line depicts the successive maximum values of ρ for which the success rate of recovery is at least 90%.

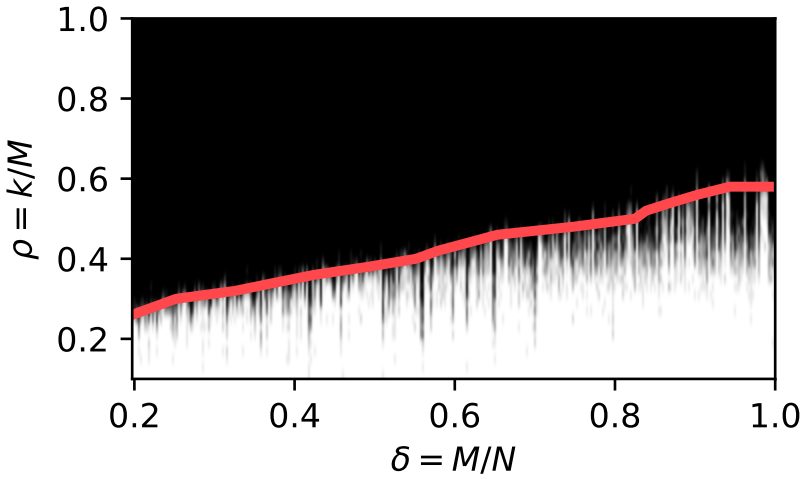
We observe two very visible phases[3]. The transition between the "success" and "failure" phases is rather sharp. This is consistent with the behavior for Gaussian matrices [56] and validates our hypothesis on phase transitions, even though the location of the transitions differs between Gaussian-based sampling and MRS. Note that the relatively high proportion of DR-MRS setups for which recovery is typically suboptimal (the "holes" below the solid line in Panel (b) of Fig. 3.2) show that the high rank property is a necessary, yet not sufficient condition for signal recovery.

Furthermore, on the $\rho - \delta$ plane, the transition is lower for DR-MRS than for QFR-MRS. For example, a signal with a given sparsity k can be well recovered by a QFR-MRS and poorly recovered by a DR-MRS, even if both MRSs have the same number of measurements M. Consider $k = 180$ non-zeros (out of $N = 1024$ samples). For QFR-MRS, $M \approx 510$ measurements

[3]Here, the term "phase" should be understood as a region in the $\rho - \delta$ plane where recovery performance is uniform, akin to a phase in a thermodynamic system. It does not refer to the angle of a periodic function.

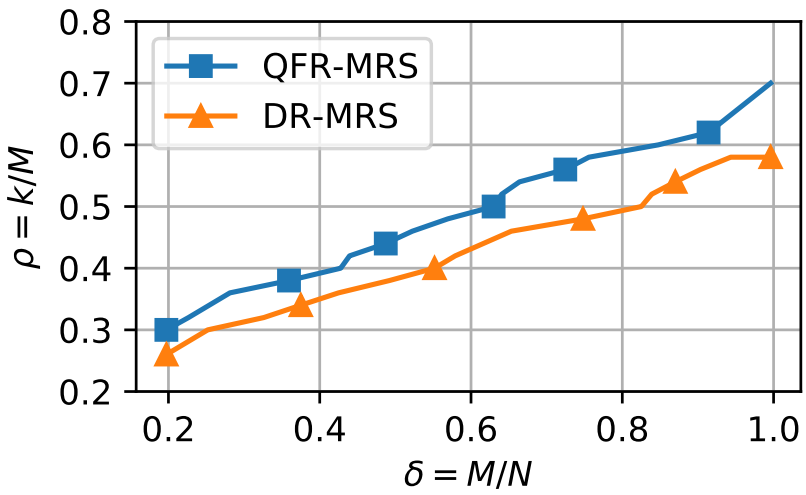


(a)

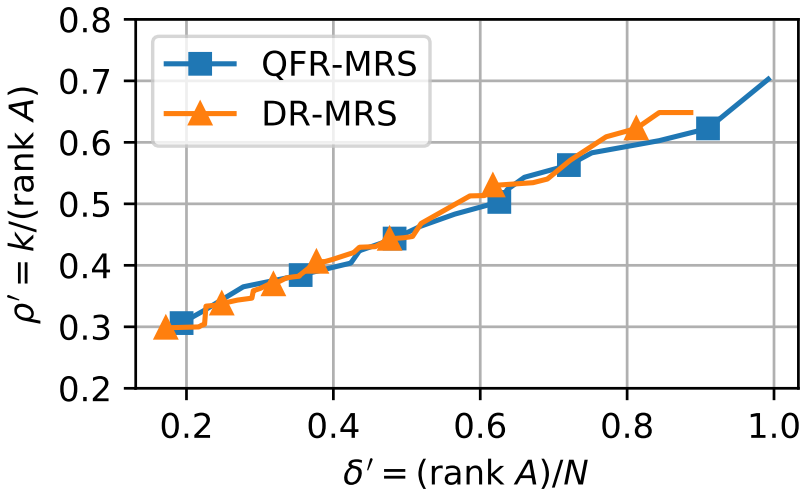


(b)

Figure 3.2: Phases for (a) QFR-MRS; (b) DR-MRS. Black phase: signal recovery is unsuccessful. White phase: signal recovery is successful. The solid line displays the transition between the two phases.



(a)



(b)

Figure 3.3: Phase transitions vs normalized (a) number of measurements; (b) matrix rank.

suffice for successful recovery ($\delta = M/N \approx 0.5$, $rho = k/M \approx 0.36$ on Fig. 3.3(a)). However, for DR-MRS, $M \approx 570$ measurements are required for successful recovery ($\delta = M/N \approx 0.56$, $rho = k/M \approx 0.33$ on Fig. 3.3(a)). In other words, in this example, successful recovery requires approximately 12% more samples for DR-MRS, compared to QFR-MRS.

In any case, this is a strong hint that the important factor here is not the number of measurements.

Transition locations depend on the MRS matrix rank Fig. 3.3 depicts the phase transitions for DR- and QFR-MRS. Panel (a) depicts the transitions on the $\rho - \delta$ plane: we can see that the transition for DR-MRS is lower that for QFR-MRS. Panel (b) represent the same transitions but on the $\rho' - \delta'$ plane, where $\rho' = \frac{k}{\text{rank } A}$ and $\delta' = \frac{\text{rank } A}{N}$. The trends for the

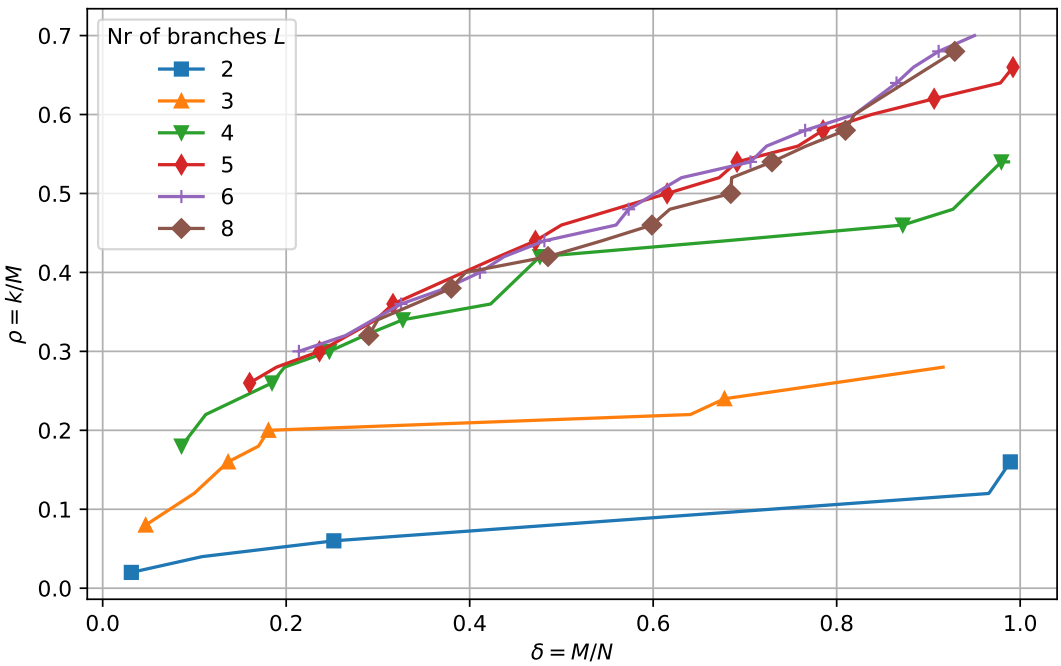


Figure 3.4: Phase transitions in the $\rho - \delta$ plane for varying number of branches L, using the CVXPY ℓ_1-minimization solver.

transitions of QFR- and DR-MRS are now extremely similar, indicating that the MRS matrix rank $\operatorname{rank}(A)$ is a more relevant factor than M when considering the recoverability of a k-sparse signal.

3.3.2.2 Asserting the impact of the number of branches L on recovery performance

As previously seen, the phase transition is lower when the sampling rates $\{M_i\}_i$ of the MRS are non-coprime. For the remainder of this chapter, we will focus on MRS systems with coprime $\{M_i\}_i$ only. Going beyond the rank indicator, we now investigate the effect of the number of branches L on recovery performance. For this purpose, we will compare recovery performance for MRS systems with the same rank, but with different values of L.

Simulation Setup The successive steps of the experiment are the same as in Section 3.3.2.1. However, the number of branches L is no longer set to 5, but varies between 2 and 8. We will see that at some point, adding more branches simply increases the implementation complexity of the MRS system without any significant impact on recovery performance.

Additionally, we only run simulations over coprime sets of sampling rates M_i, as they have shown to generally outperform non-coprime rates for a given number of measurements $M = \sum_i M_i$. Therefore, we only deal with QFR-MRS systems, refer to them as MRS, and colloquially consider that $M = \operatorname{rank} A$ as they only differ by a factor $L - 1 \ll M$.

We now focus on the phase transitions between successful recovery (RMSE under 1% at least 90% of the time) and unsuccessful recovery.

Phase transitions for various number of branches Fig. 3.4 depicts the phase transitions for noiseless k-sparse signals. Starting with $L = 2$ branches, the phase transition is rather low, meaning that only very sparse signals can be recovered correctly, no matter the total sampling rate. This suggests that a 2-branch MRS system is perhaps too simplistic for noiseless signal recovery: having only two different sampling rates does not bring in enough diversity in sampling instants, leading to low recovery performance. As new branches are added, overall recovery performance goes up, meaning that signals with a higher occupancy ratio ρ can be successfully recovered with as many different samples. However, for values of $L > 6$ branches, adding extra branches does not come with higher phase transitions. A possible explanation is that at some point (reached for $L = 6$ branches), the different sampling instants carry enough diversity to represent the sampled signal in a sufficiently rich way, so as to enable high phase transitions, and adding other sampling instants does not bring about more diversity.

Noticeably, for $L = \{3, 4\}$ branches, the values of ρ on the transition line rapidly grow with the normalized number of samples δ, but then they reach a sort of plateau (at $\delta \sim 0.19$, resp. $\delta \sim .48$ for $L = 3$ branches, resp. $L = 4$ branches) where their progression becomes limited. This effect is also present, though to a lesser extent, for $L = 5$ branches.

From $L = 6$ branches on, adding more branches do not entail better recovery. As a consequence, we recommend using $L = 6$ branches for the recovery of noiseless k-sparse signals. Note that $L = 6$ branches may only be adequate for $N = 1024$ samples. Simulations for other values of N have been inconclusive, as higher values of N entail higher-dimensional USLEs which were to computationally expensive to solve.

Note that Fig. 3.4 only represents the situation for the CVXPY solver based on ℓ_1-convex optimization used together with the MRS reduction procedure. Using other solvers may result in completely different phase transition diagrams.

Consider Fig. 3.5, where a solver based on ℓ_2-minimization or Least-Squares, is used (presented in more details in Section 3.3.3.1). The MRS reduction procedure is not applied. Furthermore, instead of setting the threshold for successful signal recovery at RMSE = 1%, it is now set to RMSE = 30%. We can see that the phase transitions are significantly shifted to the right, meaning that successful recovery with this solver is only possible for high matrix ranks – in other words, this solver has a performance inferior to CVXPY. Another observation is that changing the number of branches has no significant impact on phase transitions using this solver.

3.3.2.3 Conclusion on the recovery of noiseless k-sparse signals

From simulations conducted on k-sparse noiseless signals, we have seen that the quality of recovery increases with the number of measurements $M/\operatorname{rank} A$. However, for a given number of measurements and number of branches L, the highest recovery performance is obtained with the highest rank, which is reached when the sampling rates $\{M_i\}_i$ are coprime integers.

Furthermore, the number of branches plays a role in recovery performance. When using the right solver, recovery for a given number of measurements $M = \sum_i M_i$ is better when L is higher, up to a certain value where recovery performance reaches a peak. For signals of length

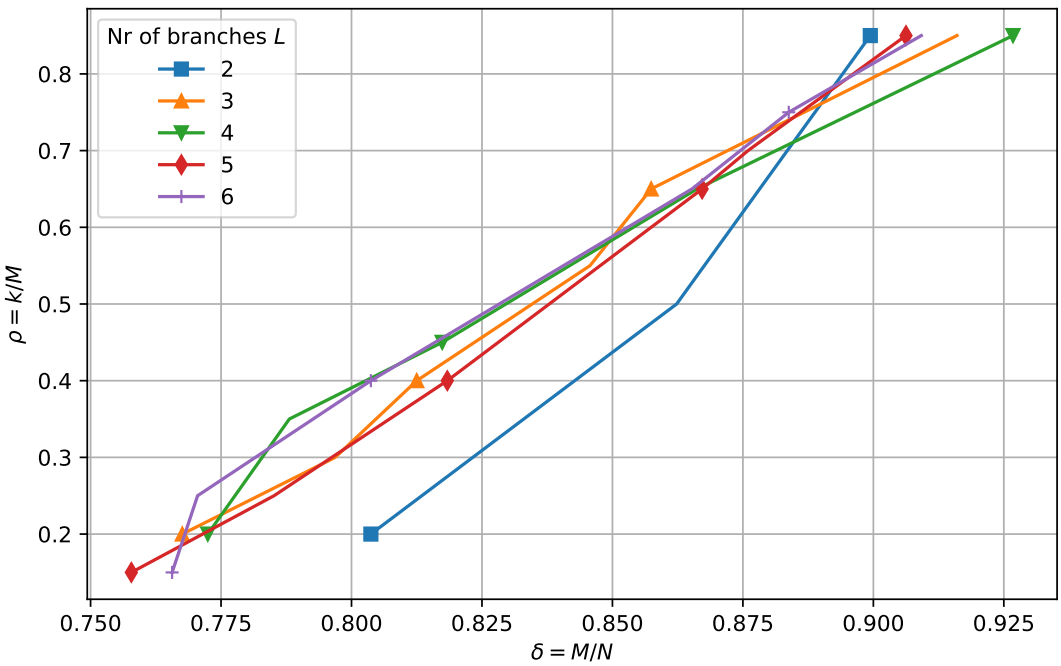


Figure 3.5: Phase transitions in the $\rho - \delta$ plane for varying number of branches L, using a Least-Squares-based solver.

$N = 1024$ samples, this tipping point is at $L = 6$ branches, and values of $L \geq 7$ branches will not entail a better recovery. Other values of N have not been studied, so the existence of a relationship between N and the optimal value of L remains an open question.

Consequently, based on simulations, we recommend having $L = 6$ coprime M_i in order to solve noiseless k-sparse signals. We now turn to the study of noisy, QPSK-based signals.

3.3.3 Application: Recovering noisy QPSK-based signals

3.3.3.1 Simulation Setup

To accomodate situations more likely to occur in telecommunications than noiseless k-sparse signals, we are now interested in the study of noisy, QPSK-based signals. Indeed, these are more representative of what an actual MRS-based sampling infrastructure would process. Four users each transmit a message modulated by a filtered QPSK on four subbands of bandwidth 1 MHz located at random positions, each distinct from one another, in a 79 MHz band. User 1's message contains useful information that we wish to recover and the other users' disposable messages consist in random bits. An Additive White Gaussian Noise with noise variance σ_w^2 corrupts the signal. more Upon reception, the signal is first sampled by an MRS system with $L = 3$ branches. Δ is set to $1024/\nu_{Nyq}$ seconds so that the block size is $N = 1024$. The values for M_i vary from 2 to 1022.

The Moore-Penrose pseudo-inverse (MPPI) of the sampling matrix A is computed and multiplied by the observation vector to yield an estimate $\hat{X}$ of the signal. Contrary to the scenario of the noiseless k-sparse signals experiment, we chose to use a basic USLE solver. Indeed, the

MPPI yields the solution of the "least-squares" (LS) problem, that is, the ℓ_2-minimization problem. Though not as good as other solvers, such as the ones based on ℓ_1-minimization or greedy optimization (see Chapter 1), the MPPI-based solver is valuable because it is fairly robust to noise, does not rely on hyperparameters and is fast. A study on solvers will be provided in Chapter 4, in which the LS-based solver will act as the baseline solver.

This scenario involves noisy signals. As a consequence, the RMSE between the transmitted signal and the received, subsampled-then-reconstructed signal bears less relevance than in the noiseless signal scenario. Therefore, we will not study the phase transitions between successful and unsatisfactory recoveries, as these are built using the aforementioned RMSE. To offer a different perspective, we propose to study recovery performance based on the BER between the transmitted message and the reconstructed one.

For this experiment, we chose to set the signal sparsity k to a fixed value. Indeed, despite the substantial number of varying parameters (sampling rates, sampling matrix rank, number of branches, E_b/N_0), we wished our analysis to remain clear and concise. In Chapter 4, different scenarios with various traffic models and occupancy ratios will be investigated.

3.3.3.2 Influence of the matrix rank on the BER

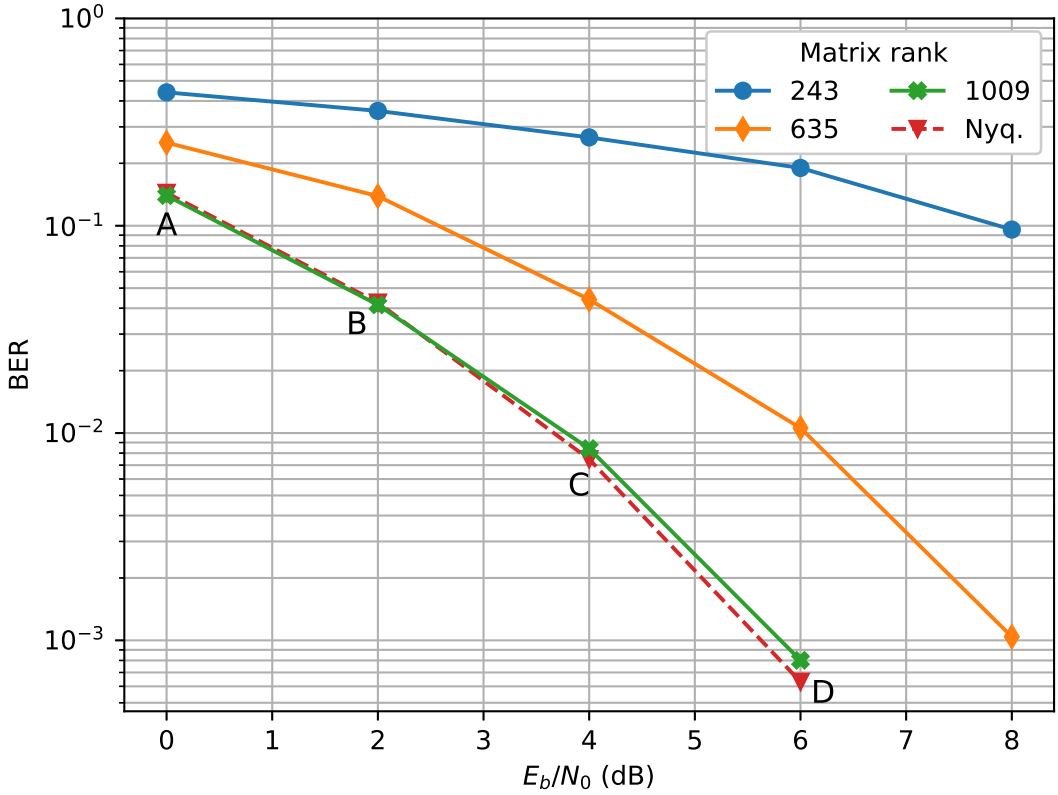
Panel (a) of Fig. 3.6 depicts the BER versus E_b/N_0 for signals sampled in different MRS setups, each corresponding to a different matrix rank. Additionally, the dashed curve is the classic BER curve for QPSK, where the Nyquist-sampled signal does not go through the MRS. For a given E_b/N_0, using an MRS with a lower matrix rank results in an increase of the BER. In particular, for MRS setups with high matrix ranks, the BER curve approaches that of the Nyquist-sampled signal.

To visualize the impact of the matrix rank on the BER more clearly, Fig. 3.6b presents the BER versus matrix rank for different values of E_b/N_0. Continuous lines represent the moving average of the BER for various sets of M_i. All curves start at a high BER, a consequence of a low-quality recovery for low matrix ranks. As the matrix rank increases, every curve approaches the Nyquist-rate BER (letters A to D). Hence, there is a trade-off between sampling rate reduction and BER minimization.

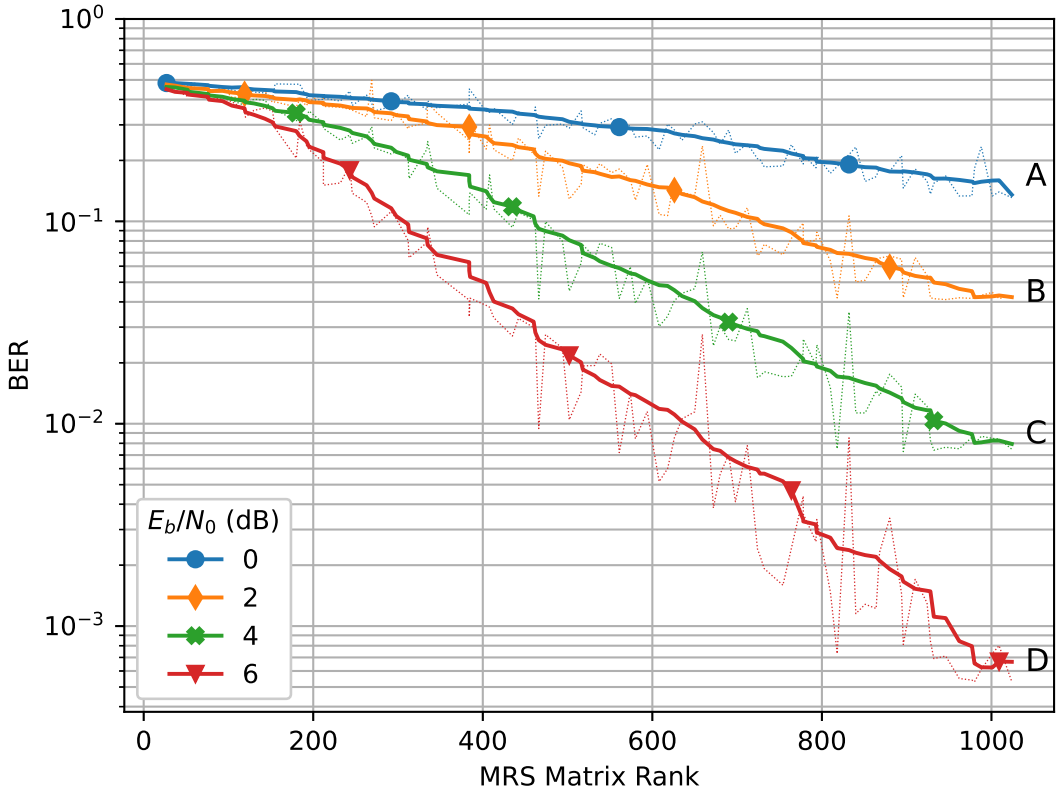
To help solve this trade-off, Fig. 3.7 can be insightful. It displays the achievable matrix rank reduction at the expense of a degraded E_b/N_0 (compared to a full-matrix-rank baseline). For example, if one can afford to lose 3.1 dB of E_b/N_0, one can reduce the matrix rank (and the overall sampling rate) by 50%. Note that the ADC literature presents trade-offs (of reduction of the sampling rate versus E_b/N_0 reduction) of similar magnitude [17]. This leads to considering the MRS as an additional source of noise, on top of other impairments undergone by the signal.

3.3.3.3 Impact of the number of branches on recovery

Panel (a) (resp. (b)) of Fig. 3.8 depicts BER vs. MRS matrix rank for $E_b/N_0 = 2$ dB (resp. $E_b/N_0 = 6$ dB) and for various numbers of branches $L \in \{2, 3, 4, 5, 6\}$. In this scenario, it appears that adding or removing branches does not have a significant impact in terms of BER between original and decoded signals. This is most likely due to the baseline USLE solver based



(a)



(b)

Figure 3.6: BER vs. (a) E_b/N_0; (b) MRS matrix rank. Matrix rank maximum value is 1024.

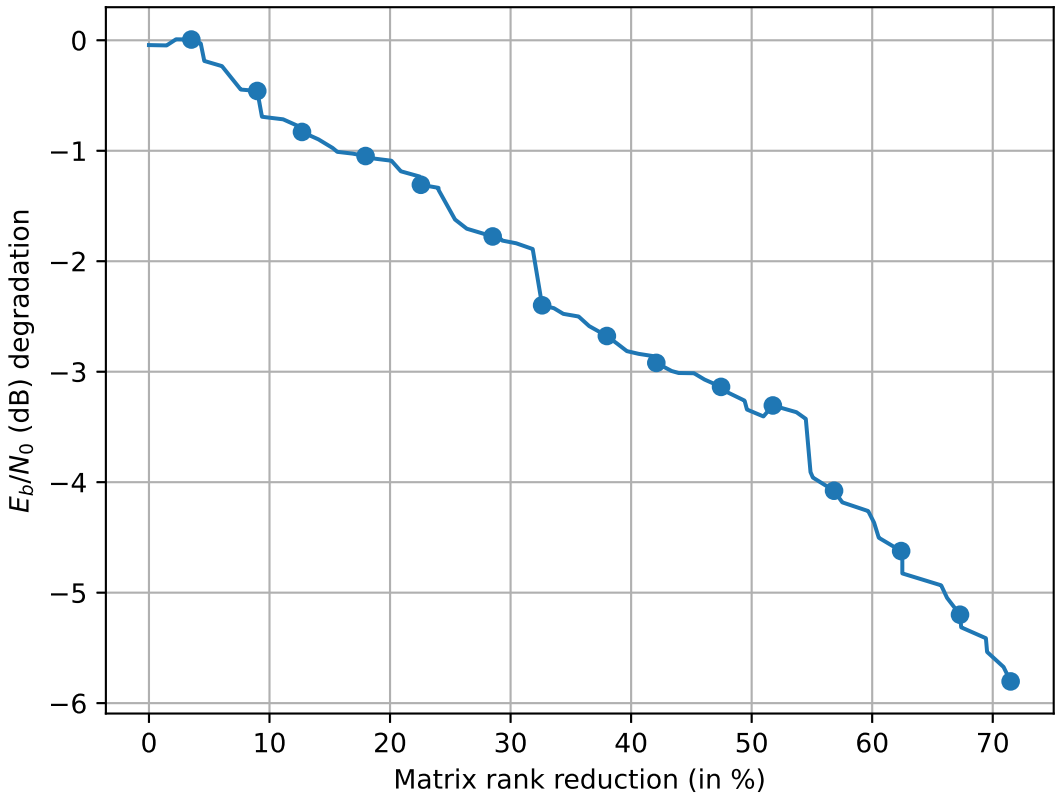


Figure 3.7: E_b/N_0 degradation vs matrix rank reduction for original $E_b/N_0 = 6$ dB.

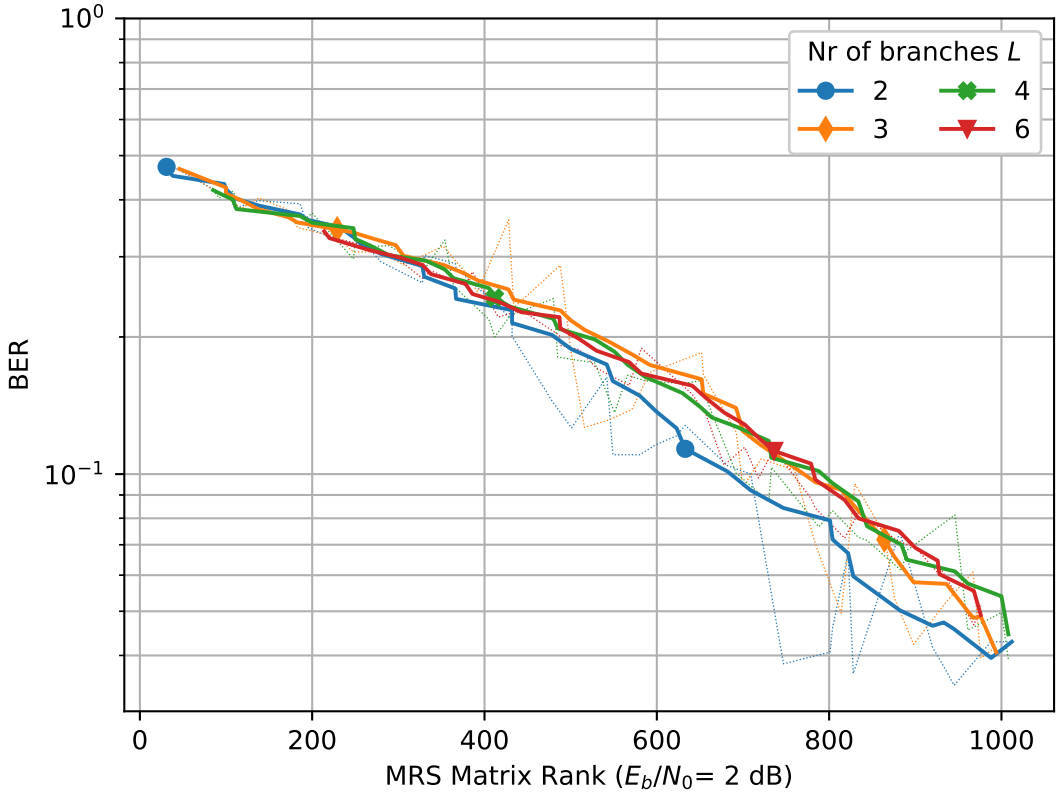
on least-squares resolution. Indeed, the same behavior is present in Fig. 3.5, which depicts phase transitions in noiseless signals using the baseline solver.

3.4 Conclusion and Perspectives

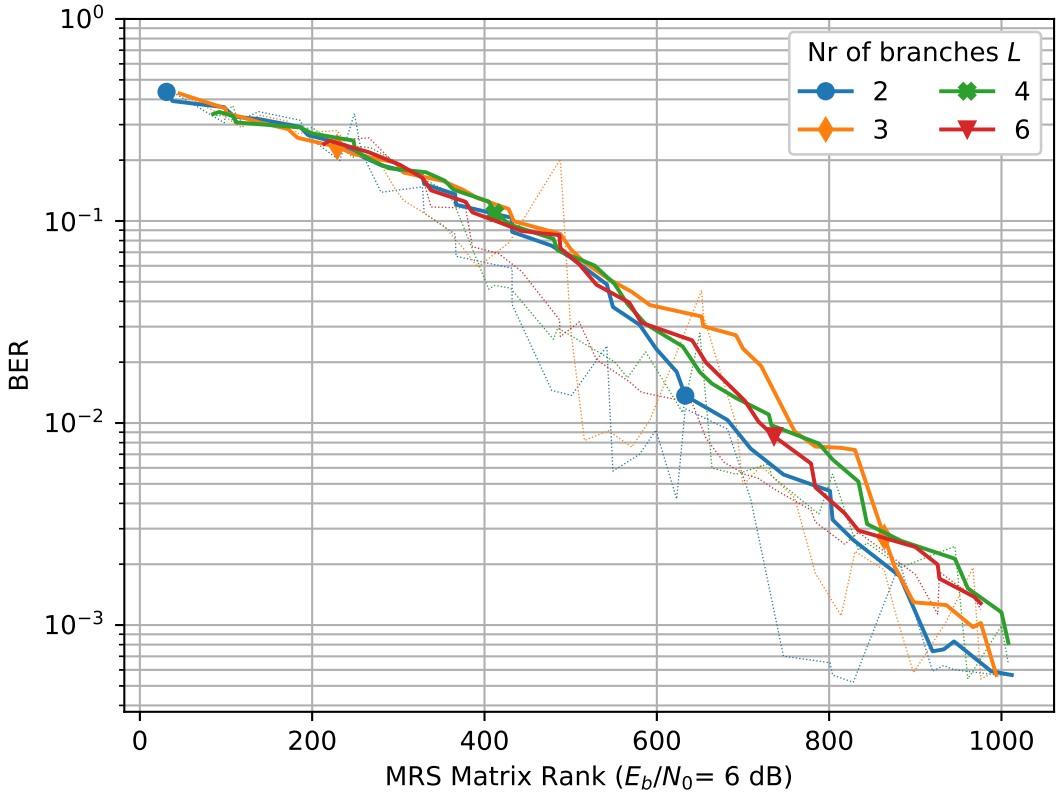
In this chapter, we have first proposed and proved a mathematical relationship between the rank of the MRS sampling matrix and the number of measurements. This relationship takes the form of an upper-bound for the matrix rank, based on the MRS sampling rates and the number of sampling branches. We have presented a simple condition to reach this upper-bound, namely, that the sampling rates be proportional to coprime integers.

Afterwards, we have linked the MRS matrix rank to recovery performance, for noiseless k-sparse signals and noisy filtered QPSK-based signals. We have seen that having coprime sampling rates results in better recovery, for a given number of measurements. Another way to improve recovery of noiseless k-sparse signals, for a given quantity of measurements, is to add more branches to the MRS - up to a certain limit where additional branches are of no use, but are the source of a higher implementation complexity. However, this effect has been documented for only one solver out of the two that we have used in this chapter: this shows how crucial to successful signal recovery the choice of the solver is. In the following chapter, we will delve into the impact of the solver on signal recovery performance.

The settings of the MRS are central to our targeted MRS-based sampling infrastructure



(a)



(b)

Figure 3.8: BER vs. MRS matrix rank with varying number of branches L for E_b/N_0 = (a) 2 dB ; (b) 6 dB. Matrix rank maximum value is 1024.

prototype. In Chapter 4, we will rely on this chapter's discussion about the settings of the MRS to adequately size the infrastructure to accomodate various types of traffic.

Chapter 4

Experimental analysis of an MRS-based sampling infrastructure for IoT traffic

4.1 Introduction

The objective of our sampling infrastructure is to acquire the information transmitted by IoT devices at all times over a wide frequency range, as proposed in Fig. 0.6 of the general introduction of this document. The number of devices and their duty cycles are unknown, so the occupancy ratio of the spectrum at a given time is unknown too. Nonetheless, our sampling infrastructure should operate in an adaptive manner, for example by acquiring fewer samples when device activity is low, while retaining the capability to detect when the conditions change so as to increase the number of collected samples. This allows the infrastructure to collect all the information transmitted by the devices, and to be able to adapt to events such as a spike in the volume of transmissions.

For this purpose, we rely on the content of the previous chapters, which aimed both at presenting the tools we developed towards acquisition of knowledge about the spectrum, effectively bringing cognitive functions to the table (Chapter 2) and at properly setting the parameters of an MRS (Chapter 3). In this chapter, we assemble these bricks towards a cognitive, adaptive, IoT-oriented sampling infrastructure based on the MRS sampling scheme.

Our proposed infrastructure is mono-site, meaning that we limit our study to one geographical site where a reception antenna is located. This can be seen as a intermediate step towards a multi-site sampling infrastructure such as the one put forward in the general introduction. The stakes and implications regarding the development of a multi-site infrastructure will be covered in the general conclusion of the document.

The reminder of this chapter is organized as follows. In Section 4.2, we present the main characteristics of our MRS-based, IoT-oriented sampling infrastructure, as well as its different parts. Elaborating on the components of our prototype, we proceed on a step-by-step basis. In Section 4.3, a IoT-sourced traffic generator is introduced. In Section 4.4, the toolkit providing

cognitive functionalities to our infrastructure, presented in Chapter 2, is tested with the MRS sampling scheme and the traffic originating from the traffic generator. Section 4.5 tackles the recovery of signals undersampled by the MRS. Section 4.6 outlines leads towards accurate infrastructure adaptivity. Finally, Section 4.7 concludes this section.

4.2 Presentation of the sampling infrastructure prototype

We now introduce our MRS-based, IoT-oriented sampling infrastructure prototype. Our infrastructure is intended as a framework for clients to deploy IoT devices and collect messages transmitted by said devices. In this regard, the prototype follows the "Infrastructure as a Service" (IaaS) paradigm.

The infrastructure, displayed in Fig. 4.1, is composed of the following blocks:

1. an analog front-end, which consists in a reception antenna and an analog preprocessing component;
2. an L-branch MRS system;
3. a K-means-based noise variance estimator (KNVE) in each branch;
4. a partial support estimator (SE) in each branch;
5. a support combinator (SC);
6. a database (DB) to store samples;
7. a signal reconstruction block.

We now review the different blocks of the infrastructure.

Analog front-end First, a wideband antenna[1] receives the transmissions sent by the IoT devices (the nature of this traffic is detailed in Section 4.3). Then, the received signal goes through usual analog components: passband filter, low-noise amplifier. Impairments provoked by these components are modeled as AWGN in our study. The output of the analog preprocessing block is a continuous signal $y(t)$. Following the notation in use in this document, it is assimilated to a discrete signal $y[n]$ sampled at the Nyquist rate (noted Y in the frequency domain).

MRS The received signal goes through L parallel branches. In the ith branch ($1 \leq i \leq L$), an ADC uniformly samples the signal at a sub-Nyquist rate to yield the sequence $z_i[n]$. The sampling rate of the ith ADC is ν_i, set so that during Δ_{block} seconds[2], M_i samples are acquired. N is the corresponding number of samples for a Nyquist-rate acquired signal. The sequence $z_i[n]$ undergoes a DFT, so the outcome of this step at the ith branch is a M_i-sized complex vector noted Z_i.

[1]The antenna is wideband with regard to the underlying signals.

[2]See Section 4.3.1 for an introduction to our block-processing paradigm and to the duration Δ_{block}.

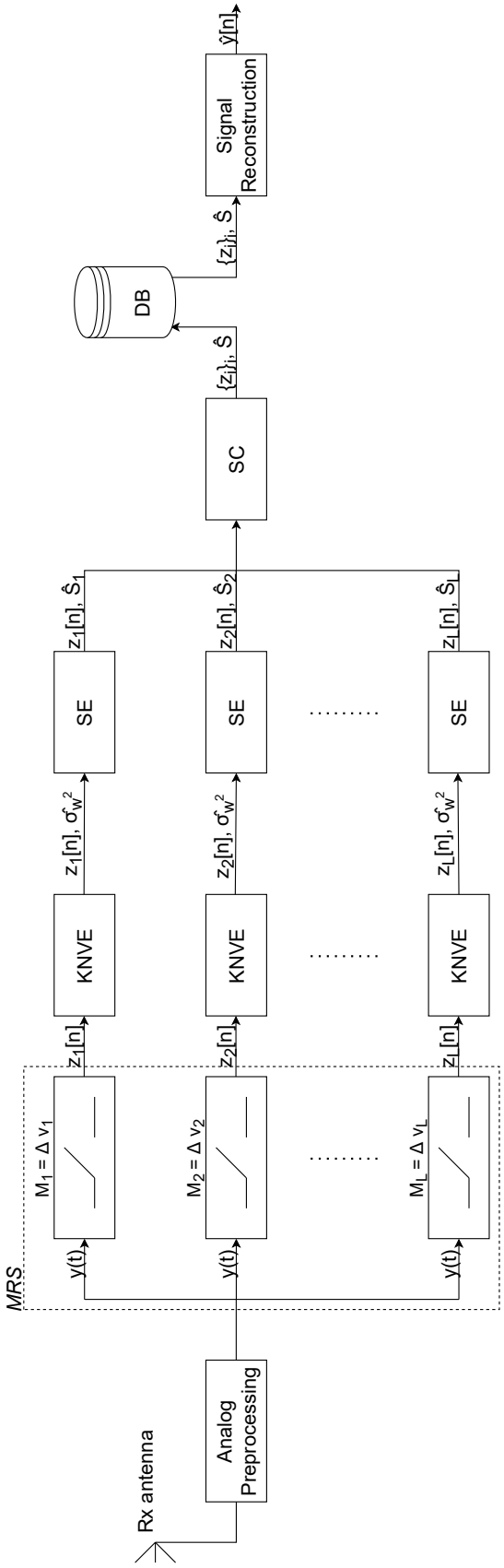


Figure 4.1: Schematic describing our proposed MRS-based sampling infrastructure.

KNVE In each branch, the variance of the additive white noise is estimated to yield $\hat{\sigma}_w^2$ (index i is omitted because the noise variance estimates of the different branches are never processed jointly). The $\{Z_i\}$ are also propagated.

SE Based on the vector Z_i and noise variance estimate $\hat{\sigma}_w^2$, a partial support estimation is carried out in each branch to yield $\hat{S}_i$, a M_i-sized boolean vector. This support estimate is called "partial" because it reflects the support as seen in an undersampled signal in one branch (due to undersampling, the spectral components of the signal of interest are folded onto different locations of the frequency spectrum). Again, the Z_i are propagated for future processing.

SC Combining the partial support estimates from all L parallel branches allows to estimate the frequency support of the received signal $y[n]$ as if it were sampled at the Nyquist rate. This support estimate, noted $\hat{S}$, is a boolean vector of size N.

DB The signal samples $\{Z_i\}$ and support estimate $\hat{S}$ are stored in a database. The samples and support estimate can be stored for a given duration that depends on the options of the IaaS contract. For the samples, there are two possible storage modes: full storage and information-only storage. In the first storage mode, all samples $\{Z_i\}$ are stored in base. In the second storage mode, only samples identified to contain signal of interest (through the support estimate $\hat{S}$) are stored in base. The choice of the storage mode is intrinsically linked to the choice of the solver in the Signal Reconstruction block.

Signal Reconstruction According to CS literature (reviewed in Chapter 1), an underdetermined system of linear equations (USLE) is associated to every CS sampling scheme. Solving this USLE for Y is called signal reconstruction, or signal recovery. It is carried out on an on-demand basis upon a user request. Several solvers can be used for this purpose. Some solvers take advantage of the reduction procedure of the MRS, originally introduced in [20], through which the dimensionality of the USLE to solve is reduced using the knowledge brought by the support estimate $\hat{S}$. In this case, the information-only storage mode of the DB can be favored. However, other solvers require all samples for signal reconstruction, in which case it is necessary to use the full storage mode of the DB.

After presenting the different blocks of the proposed sampling infrastructure, we dive deeper into every block (except the analog preprocessing and DB blocks). But first, we introduce our traffic generator, aimed at accurately simulating IoT-like transmissions.

4.3 The IoT-oriented traffic generator

Our infrastructure is aimed at collecting samples of signals corresponding to IoT communications. In order to test our infrastructure prototype, we have developed a traffic generator to simulate messages transmitted by IoT devices. As its name suggests, the traffic generator generates a traffic, which is essentially a (typically wideband) signal containing data components. The role of the traffic generator is to create and modulate messages that encapsulate data, to

translate them into IF/RF and finally, to transmit them in a channel. We have opted for a traffic generator that is completely simulative and do not actually transmit a traffic over the air using antennas. As a consequence, the generated traffic is discrete by nature.

This traffic generator is characterized by several parameters, that we now introduce and discuss while keeping in mind the IoT context.

4.3.1 Parameters of the traffic generator

The traffic generator is characterized by the following three parameters: bandwidth, duration, and channel model and block size. We now review the choice of parameters we have made for the traffic generator.

Total observation bandwidth The aim for the generated traffic is to emulate a realistic sparse and multi-band signal. For this purpose, we have chosen a channel bandwidth $BW = 100$ MHz. This is the same bandwidth as the 2.4 GHz ISM band. When the BW is set to higher values, a resolution problem arises. Indeed, for a given block size (the number of samples processed at once, in a batch fashion), the sparse signal components are represented by fewer samples in the frequency domain as the total bandwidth increases. As the block size remains the same and the total bandwidth increases, the sampling duration and therefore the overall frequency resolution decreases. This can be problematic, as too low resolutions notably prevent correct support estimation. A solution is to increase the block size, as it increases the frequency resolution. However, this comes at the expense of higher computational requirements: as the block size increases, so does the USLE size.

Duration The duration is useful to set the total number of samples in the generated traffic. However, it corresponds to no physical quantity. It is more to be seen as a parameter for our batch-processing chain, in which all the traffic is generated first, than as an intrinsic parameter of the traffic generator having a significant influence on an implemented stream-processing chain. We have set it to $\Delta = 500$ ms. As a consequence, the maximum[3] number of samples in a traffic is $\Delta \times BW = 5 \times 10^7$ complex samples.

Channel model An actual sampling infrastructure receives signals subject to fading. Fading is usually classified as either short-scale or long-scale, depending on the characteristics of the fades. Long-scale fading is mainly caused by shadowing or path loss and is considered not to be time-varying. Short-scale fading is principally due to multi-path fading and scattering, and is considered to be time-varying. We made the decision not to account for long-scale fading, as it is mainly due to the geographical configuration of the reception site and its surroundings. The rationale behind this decision is that if transmissions from a device undergo shadowing, there is not much a mono-site sampling infrastructure can do about it.[4] We also consider that the channel impulse response is frequency-invariant. This strong assumption is mainly formulated

[3]This maximum number of samples corresponds to Nyquist-rate sampling.

[4]On the contrary, this is typically the sort of impairment that a multi-site infrastructure could help mitigate.

to keep the study tractable. If a mechanism towards estimation of the channel impulse response is implemented, its outcome can be used to equalize reconstructed Nyquist-rate samples $\hat{Y}$ for further information retrieval, but this is out of the scope of our study.

Following these assumptions, we implement the Rayleigh channel model to account for small-scale fading. Additionally, we add white Gaussian noise to account for thermal noise and impairments caused by the analog preprocessing component. As a consequence, the frequency-domain representation of the received signal is:

$$Y = HX + W, \tag{4.1}$$

where Y is the received signal as processed by the MRS (impairments due to the analog preprocessing component are comprised even if they technically occur after reception by the Rx antenna), H is the (frequency-flat) multiplicative Rayleigh channel, X is the sum of the signal components (each signal component represents an IoT transmission) and W is the AWGN.

Block size The block size N is the number of Nyquist-rate samples processed together in one batch. A block represents the acquisition of a signal for a duration of $\Delta_{block} = N/BW$. The block size is to be set carefully: a low N entails a smaller acquisition duration and a lower frequency resolution, but a high N results in a bigger and more computationally expensive USLE. It is to note that some processing steps, *e.g.* noise variance estimation or even support estimation, can take place over several blocks, as they are less computationally demanding than USLE resolution. In general, we will use $N = 2000$ samples, as it is a reasonable trade-off between the size of the underdetermined problem to solve and the frequency resolution/acquisition duration. At $BW = 100$ MHz, we have $\Delta_{block} = 20$ µs.

Medium access We consider an extremely basic medium access control scheme in which the devices transmit at random frequencies and instants, without any prior listening. Interference is assumed to be mitigated by the sporadicity of the communications in the time and frequency domains. Transmission power is however controlled: each device transmits their message with the same power.

After discussing the parameters of the traffic generator, we now turn to the presentation of the characteristics of the signal components representing IoT messages.

4.3.2 Modulation types

We focus on three IoT standards that are popular at the time of writing this manuscript:

- LoRaWan, using chirp-based spread spectrum modulation;
- NB-IoT, using OFDM;
- eMTC (LTE Cat. M2), also using OFDM.

These three standards all have their modulations, failsafe mechanisms and other characteristics *e.g.* payload size, typical frame duration, and so on. Our platform aims to be technology-agnostic, so as to allow an easier introduction of new actors with their own modulations and spectrum occupancy characteristics. As a consequence, we do not focus on specific modulations, but rather on some characteristics of these three standards, namely their typical bandwidth and frame duration.

We create frames that have the same bandwidth than the three aforementioned standards, with an on-air duration corresponding to a payload of 100 bytes, a size representative of a variety of IoT applications, such as sensors communications for the smart city. To simplify the generation and analysis of the frames, we mimic them using the filtered QPSK modulation, with bandwidths and durations set to match the payload size of 100 bytes for the actual modulation. Due to the modulation being different (filtered QPSK instead of whatever the actual modulation of the standard is), the payload size may not actually be 100 bytes, but is instead linked to the bitrate and duration of the QPSK frame. Nonetheless, we are aware that the actual waveforms may have a different behavior than QPSK on the different steps of our sampling infrastructure. Our payloads consist in randomly generated ASCII messages, with no regard for traditionally encapsulated overhead *e.g.* receiver address or frame check sequences, as this is a subject beyond the recovery of Nyquist-rate samples from a signal acquired below the Nyquist rate. The nature of the frames for the different standards is now discussed.

LoRaWan-like frame LoRaWan frames have three possible bandwidths of 125 kHz, 250 kHz and 500 kHz (only 125 kHz and 250 kHz in Europe). The duration for a 100-byte payload varies with the bandwidth used and the spreading factor, a parameter varying between 7 and 12 whose effect is to improve the communication range of LoRaWan at the expense of a higher energy consumption and a longer duration. For a 100-byte payload, the duration varies between 95 ms (SF7, bandwidth = 125 kHz) and 615 ms (SF9, bandwidth = 250 kHz - it is to note that higher spreading factors are not available with this payload size as per the LoRaWan specs). For our LoRaWan-like frame, we will use a duration of 95 ms with a 250 kHz bandwidth. This bandwidth represents 0.25% of the total bandwidth of the generated traffic, *a priori* in contradiction to the compacity hypothesis of the support estimator presented in Chapter 2.

NB-IoT-like frame NB-IoT uses a 200 kHz bandwidth. At 40 kbits/s (the uplink peak rate is 66.9 kbps for a single-tone system), the duration for a 100-byte payload is approximately 20 ms. As a consequence, we use a bandwidth of 200 kHz and an duration of 20 ms. This bandwidth represents 0.2% of the total bandwidth of the generated traffic, again in contradiction to the compacity hypothesis of the support estimator presented in Chapter 2.

eMTC-like frame The LTE Cat M2 specification for eMTC proposes a 5 MHz bandwidth, representing a 5% fraction of the total bandwidth of the generated traffic. At a bitrate of 4 Mbps (the peak rate for uplink communications is around 7 Mbps), the duration for a 100-byte payload is approximately 200 µs.

Other frame types In this study, we have left out other popular standards, such as Sigfox, which is based on Ultra Narrow-Band (UNB) communications. With its 100 Hz bandwidth, a Sigfox packet represents a millionth of the total bandwidth, which raises concerns about frequency resolution notably for the support estimation step. However, nothing inherently prevents our proposed sampling infrastructure to process UNB transmissions. To do so, two possible adjustments the infrastructure could implement are the following. First, the KNVE and SE would probably have to operate without the smoothing window step, as such a narrow signal component would likely not be spread over several frequency samples. Second, in an effort to increase the frequency resolution for the KNVE and SE to have an acceptable performance, the input to these components should have a longer duration. However, the resulting loss in time resolution can be detrimental to other types of modulations. One way to mitigate this issue is by having two (or more) acquisition durations targeted at the various bandwidth and duration ranges of the different modulation types.

The characteristics of the frames transmitted by our traffic generator are summarized in Table 4.1.

Frame type	LoRaWan-like	NB-IoT-like	eMTC-like
Bandwidth	250 kHz	200 kHz	5 MHz
Duration	95 ms	20 ms	200 µs

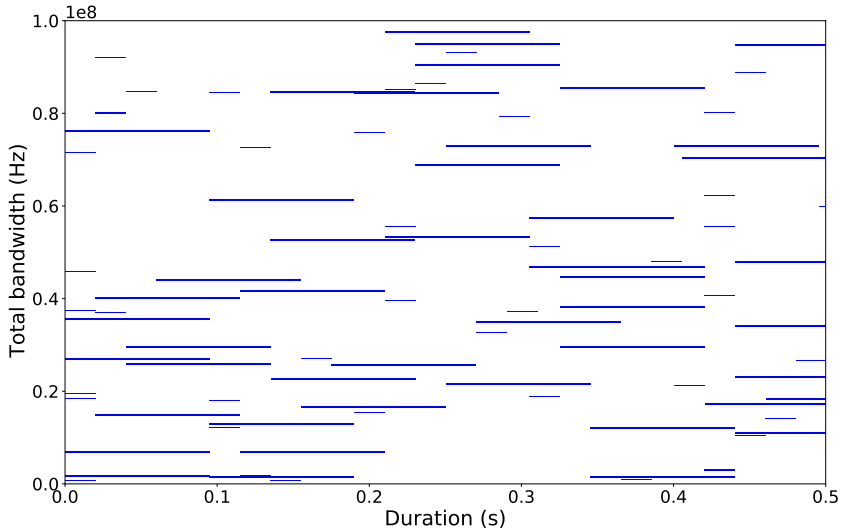
Table 4.1: Characteristics of our QPSK-based frames used to simulate the behavior of IoT communications.

We now introduce the different scenarios for the generated traffic, mainly characterized by its instantaneous occupancy ratio.

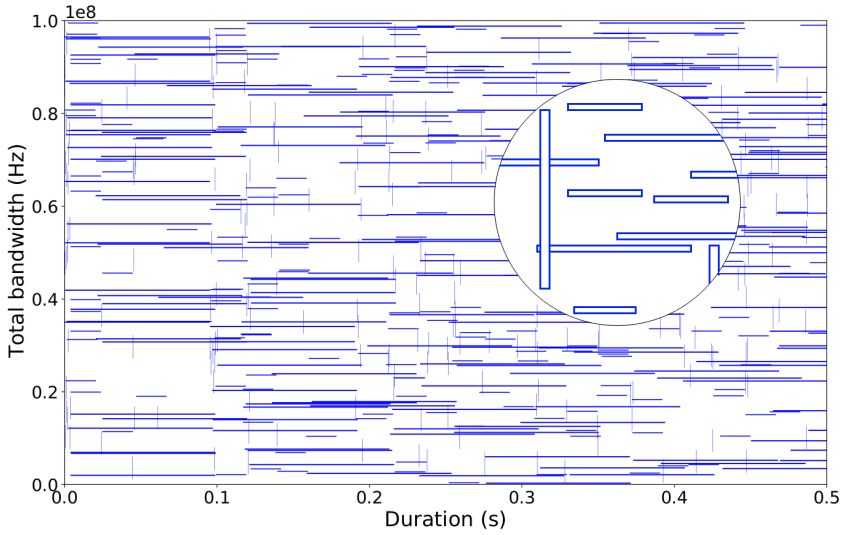
4.3.3 Traffic types

To characterize the performance of our sampling infrastructure prototype, we test it for several scenarios of traffic volume. The difference between the scenarios is the level of spectral occupancy or spectral occupancy ratio. We chose occupancy ratios that seemed reasonable and that could correspond to realistic use cases. The three used scenarios are the following:

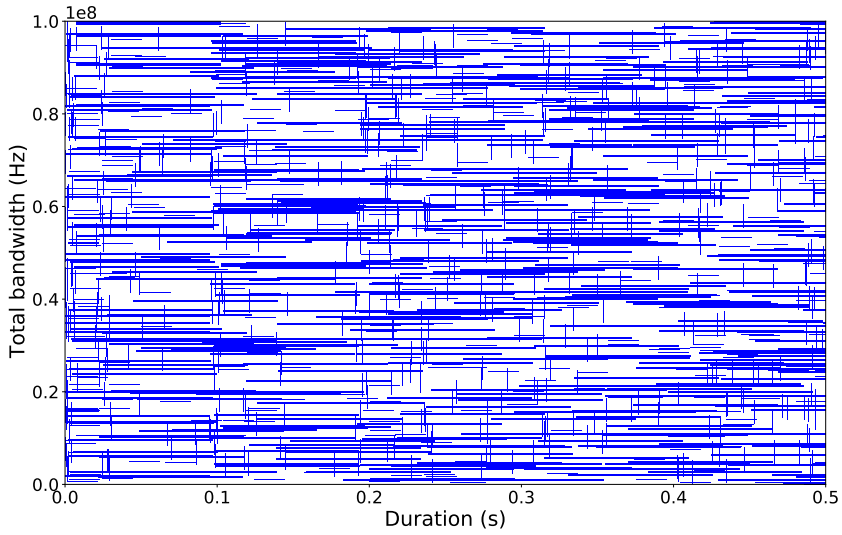
- Scenario 1: Low-volume traffic, corresponding to an area with a limited amount of deployed IoT devices which seldom communicate. The occupancy ratio is between 2-3%;
- Scenario 2: Medium-volume traffic, with more devices that communicate more often. The occupancy ratio corresponding to this scenario is 8-12%;
- Scenario 3: High-volume traffic, where there are many devices, possibly with high duty cycles. This can also correspond to an exceptional situation where there is a burst of activity for a great quantity of devices. The corresponding occupancy ratio is 22-28%.



(a)



(b)



(c)

Figure 4.2: 2D time-frequency traffic visualizations for average occupancy ratios of (a) 2.5%; (b) 10%; (c) 25%.

	Traffic scenario 1	Traffic scenario 2	Traffic scenario 3
High SNR regime	3 dB	9 dB	12 dB
Medium SNR regime	−3 dB	3 dB	6 dB
Low SNR regime	−9 dB	−3 dB	0 dB

Table 4.2: Total SNR for the different SNR regimes and traffic scenarios.

Traffic scenarios with even higher occupancy ratios correspond to signals that we have found not to be able to be processed satisfactorily by our sampling infrastructure, as they exhibit too little sparsity considering our assumptions and infrastructure setup. These are therefore excluded from our study. Fig. 4.2 displays examples of generated traffics in the 2D time-frequency plane. Each rectangle represents a frame: the LoRaWan-like frames are the long horizontal ones, the NB-IoT-like frames are the short horizontal ones and the eMTC-like frames are the vertical ones.

From Section 4.4 on, we consider the different traffic scenarios in three different SNR regimes (high, medium and low). The impactful factor is the in-band SNR, defined as the energy of the signal of interest divided by the energy of the noise *in the support of the signal of interest*. However, in this study, we consider the total SNR (defined as the energy of the signal of interest divided by the energy of the noise *in the entire band of observation*), which depends on the in-band SNR and on the occupancy ratio. The correspondance between the SNR regime, the traffic scenario and the (total) SNR is found in Table 4.2.

We have defined the conditions in which our simulations take place. Now, we turn to the experimental validation of our preprocessing methods put forward in Chapter 2, namely, the noise variance estimator and the signal support estimator.

4.4 Validation of the toolkit introduced in Chapter 2

4.4.1 Introduction

In Chapter 2, we have proposed tools to determine the frequency-domain support of sparse multi-band signals, a step towards simplified USLE resolution.

Early in this section, we present some practical aspects that need to be considered when it comes to using these tools, namely the K-means-based noise variance estimator (KNVE) and support estimator (SE). We then turn to the analysis of the performance of the KNVE and SE as part of our proposed sampling infrastructure.

Another subject tackled in this section is the recovery of the Nyquist-rate support from the partial support estimates acquired at every MRS branch. The SE's outcome represent the support for the signal sampled by one branch, which is different from the support of the Nyquist-rate sample sequence due to undersampling - this is the very principle of the MRS. To recover the support estimate for the Nyquist-rate sequence, we apply a procedure originally introduced by Fleyer *et al.* in [20]. In Section 4.4.5, we will characterize this procedure and identify its strengths and weaknesses.

The structure of the remainder of this section is as follows. First, we provide a discussion about practical considerations to keep in mind when using the KNVE and SE on signals generated by our IoT traffic generator. Second, we indicate suitable parameters to operate the KNVE in our sampling infrastructure for the different traffic scenarios. Third, we study the performance of the SE in each MRS branch. Fourth, we focus on the support combination procedure to recover the Nyquist-rate support estimate from the support estimates taken from each MRS branches.

4.4.2 Practical considerations when using our toolkit

Several aspects need to be considered when it comes to using the KNVE and SE as part of our sampling infrastructure. These are now discussed.

Low compacity of the signal components First, the nature of the traffic is less favorable than what the KNVE and SE have been tested for in Chapter 2. Indeed, signal components presented in Section 4.3.2, especially the LoRaWan-like frame and the NB-IoT-like frame, have a very little compacity in that they occupy extremely small fractions of the spectrum. This affects the windowing step of both the KNVE and SE.

Change in occupancy ratio when undersampling Second, sampling using the MRS entails a change in spectrum occupancy in the different MRS branches. To comprehend this, remember the spectrum occupancy ratios of the different traffic scenarios introduced in Section 4.3.3 are given for Nyquist-rate sampling. As the traffic undergoes undersampling through each MRS branch, every signal component is aliased onto a smaller total bandwidth. Consequently, the occupancy ratio of the sample sequence acquired at each MRS branch is higher than that of the Nyquist-rate-acquired sequence. As a rule of thumb, for sparse signals, if the sampling rate in branch i is a fraction $\delta_i = M_i/N$ of the Nyquist rate, the occupancy rate is multiplied by δ_i. Note that this rule of thumb does not take into account different signal components that are undersampled onto the same frequencies. The repercussion is that a signal whose Nyquist-rate representation is sparse does not necessarily have a sparse representation when undersampled by an MRS branch. This is a concern as the KNVE and SE rely on a hypothesis of sparsity, yet are applied to the sequences sampled at each MRS branch.

In this section, we will present adjustments to mitigate and overcome these issues. These solutions principally consist in increasing the number of samples on which the KNVE and SE are applied. This is made possible both by the relatively low complexities of the two methods and by the relatively high time resolution with regard to the duration of the various frame types considered.

We now turn to the analysis of the performance of the KNVE as part of our proposed sampling infrastructure.

4.4.3 Using the K-means-based noise variance estimator in our sampling infrastructure

After the received signal is sampled using the MRS, the next step in the signal processing chain is the KNVE. The KNVE is applied to DFT Z_i of the sample sequence $z_i[n]$ acquired in one branch. The idea is to use the output of the KNVE, that is, the estimated noise variance, as an input to the energy-based SE. As a reminder, the KNVE is based on separating noise-only values of the received sample sequence from signal-plus-noise values using K-means.

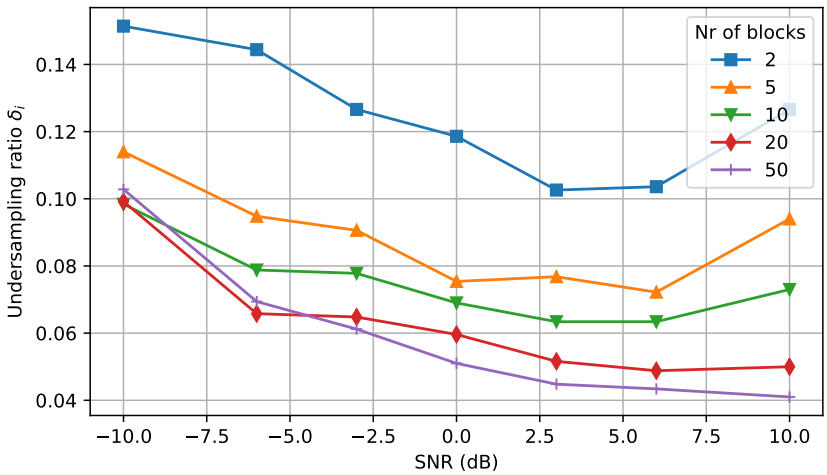
The noise-only values consist of additive noise and are not subject to Rayleigh fading, contrary to the signal-plus-noise values which are affected by Rayleigh fading. Therefore, even if the KNVE was put forward in the context of the AWGN channel in Chapter 2, the impact of the Rayleigh fading channel should be moderate.

As mentioned in Section 4.4.1, the occupancy ratio of the signal undersampled in an MRS branch is higher than its Nyquist-rate counterpart. Indeed, all the original signal components are folded into a smaller bandwidth. For low sampling rates, the resulting occupancy ratio can be fairly high, encroaching on the KNVE hypothesis of a sparse input signal. Furthermore, the signal components considered in our application have a low compacity, in the sense that they consist in many narrowband signals instead of a few "wideband" signals. This means that considered signals typically fail to meet KNVE's second hypothesis of compacity. Luckily, our tests have demonstrated the robustness of the KNVE even if the two hypotheses are not completely satisfied.

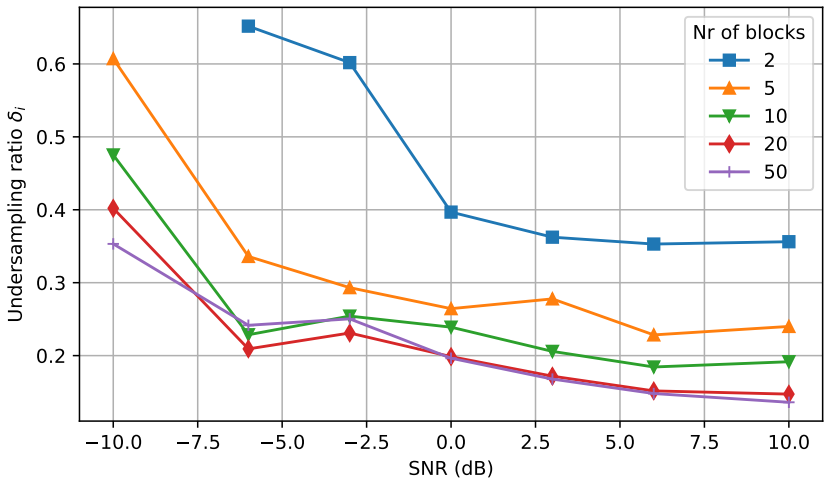
To analyse the performance of the KNVE for our application, the following experiment is conducted. For a given traffic scenario, the received signal is sampled at some rate $M_i = N\delta_i$ (δ_i is called the undersampling ratio at branch i) for durations varying from 40 µs to 1 ms. The noise variance is estimated (the estimate is noted $\hat{\sigma}_w^2$) and compared to the actual noise variance σ_w^2 of the received signal. The estimation is considered successful when the relative error $RE = |\hat{\sigma}_w^2 - \sigma_w^2|/\sigma_w^2$ is below 5%. This simulation is carried out on a signal which went through a Rayleigh fading channel and for SNRs varying from -10 dB to 10 dB (except for Scenario 3 for which results for SNR of -6 dB and below are inconclusive). The simulations are averaged over 100 runs.

Fig. 4.3 depicts the average optimal (lowest) undersampling ratios for which KNVE makes a successful estimation, as a function of the SNR. Each of the three scenarios is represented by a subfigure. Since each block of samples lasts 20 µs, the "number of blocks" parameter (n_{blocks}) measures the acquisition time of the sequence of samples on which the KNVE was applied. It is also a measure of the number of samples on which the KNVE was applied: this number of samples is $n_{blocks}\delta_i N$. From these simulation results, some observations are summarized as follows.

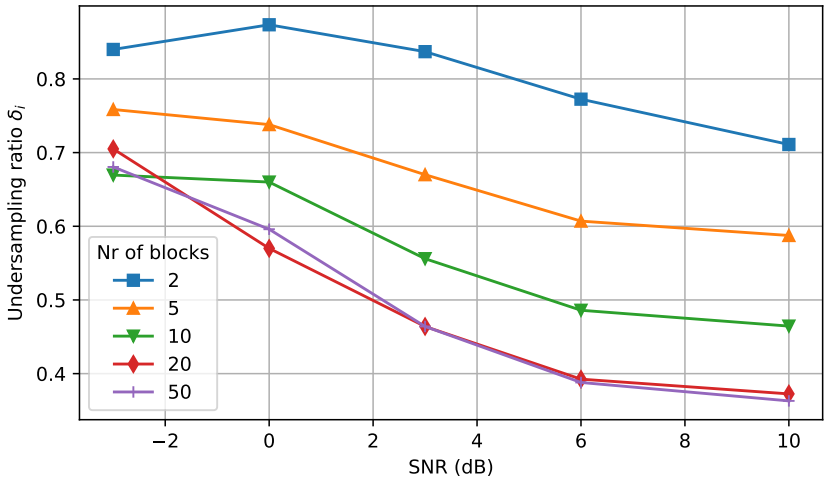
The lower the traffic volume, the lower the sampling rates. As the average occupancy ratio increases, so does the requirement for higher sampling rates. While correct estimation can be achieved with a sampling rate as low as 4% of the Nyquist rate for Scenario 1, a higher sampling rate of 36% Nyquist is required in Scenario 3. Noticeably, between the different



(a) Scenario 1: 2.5% average occupancy ratio.



(b) Scenario 2: 10% average occupancy ratio.



(c) Scenario 3: 25% average occupancy ratio.

Figure 4.3: Optimal undersampling ratios δ_i for successful noise variance estimation with the KNVE versus SNR, for different sampling durations (represented by the number of 20-µs blocks) and different traffic scenarios.

scenarios, the scale (for $n_{blocks} = 50$ and SNR = 10 dB) between the optimal sampling rate and the average occupancy ratio is approximately constant, at ~ 1.5. This means that for high SNRs, a rule of thumb to find the lowest suitable sampling rate for the KNVE is to multiply the average occupancy ratio by $1.5N$.

The longer the acquisition duration, the better. Increasing the number of blocks on which the KNVE is applied lowers the requirements on sampling rates, for every scenario. For instance, in Scenario 1, a sampling rate of 12% Nyquist is required when the decision is taken on $n_{blocks} = 2$ blocks, and this requirement drops by half when raising n_{blocks} tenfold. While this hike in sampling duration might seem large, we remind the reader that the KNVE is scalable due to a linear complexity, and the total duration of the acquisition (of 1 millisecond for $n_{blocks} = 50$ blocks) remains low compared to the rate at which the noise variance changes. However, it is to note that the law of diminishing returns applies here, and at some point, further expansion of the sampling duration does not come with lower sampling rates requirements. For example, setting $n_{blocks} = 50$ blocks gives about the same performance as $n_{blocks} = 20$ blocks.

For lower SNRs, correct estimation requires higher sampling rates. For lower SNRs, it is important to accurately estimate the noise variance, so as to precisely set the threshold of the SE. However, when confronted with lower SNRs, the KNVE has more difficulty separating noise-only values from signal-plus-noise values, and can use a little help in the form of a relaxed sampling rate requirement, which allows for a higher fraction of noise-only values.

Note that in Scenario 1, the surge observed for SNR = 10 dB can be explained by the fact that the actual noise variance is possibly extremely small, drastically increasing the relative error RE on the estimation despite a low absolute error (for example due to spectrum leakage). However, it is not extremely worrysome: even if off by, say, 10%, the noise variance estimation will be sufficient for the SE for such a high SNR.

After studying the conditions in which the KNVE operates, we turn to the parameterization of the SE.

4.4.4 Signal support estimation in each MRS branch

The stakes regarding the use of the frequency-domain support estimator (SE) in our sampling infrastructure are now reminded. Because we do not know the locations of the signal components in the frequency spectrum in advance, we have to sample at a higher rate than what could be achieved if these locations were known beforehand. In other words, our sampling infrastructure typically collects samples that are *not* strictly necessary for recovery of signal components. In an effort to minimize the amount of samples to be processed and stored in base, we strive to discard samples that do not convey relevant information. For this purpose, the SE can be very helpful as it separates a frequency-domain sparse signal into (frequency-domain) information-containing samples on the one hand, and (frequency-domain) noise-only samples on the other hand.

Given that our sampling infrastructure relies on sub-Nyquist sampling, a naive approach would be to first recover the Nyquist-rate samples from the sub-Nyquist samples using CS tools, then from the recovered Nyquist-rate samples, to identify the frequency-domain support of the received signal using the SE to discard useless samples. However, the MRS scheme, based on undersampling, allows for a more clever use of the SE, where intermediary recovery of the Nyquist-rate samples is not required. Consider the different branches of the MRS, in which each signal is undersampled. At each branch, the acquired sample sequence contains the same signal components, except that they are aliased onto different frequencies from one branch to another. It is therefore possible to perform support estimation on the sample sequence acquired at each branch Z_i: the resulting estimate $\hat{S}_i$ then represents the frequency-domain support of the aliases of the signal components. As we will see in Section 4.4.5, it is possible to combine these so-called partial support estimates $\{\hat{S}_i\}$ to recover the frequency-domain support of the signal $\hat{S}$ as if it were sampled at the Nyquist rate, using computations of a far lower complexity than CS tools for USLE resolution.

In this section, we focus on the computation of partial support estimates $\{\hat{S}_i\}$, that is, on sample sequences acquired at each MRS branch. Each support estimate $\hat{S}_i$ is a M_i-sized boolean vector whose jth element is 1 if signal-of-interest is present at the jth position of the sampled signal's DFT and 0 otherwise. For the combination of these partial support estimates into the so-called total support estimate, see Section 4.4.5.

4.4.4.1 Parameters of the SE

A few parameters of the SE are now discussed in light of the context in which it is applied.

Type I/II error trade-off The SE relies on binary hypothesis testing, and the choice of the detection threshold involves an arbitration between type I errors (false positives) and type II errors (false negatives). In our application, a type I error results in a noise-only sample being labeled as containing information, while a type II error discards an information-containing sample as noise-only. Type I errors call for needless computation and storage resources, and should generally be avoided; but type II errors result in a net loss of information, defeating the original purpose of the sampling infrastructure. Therefore, we seek to minimize type II errors by setting a requirement of $P_D \geq 0.99$ on the probability of detection. Following this requirement, we wish to minimize the probability of false alarms P_{FA} which measures the proportion of type I errors.

The resolution issue and the SE window length The frequency resolution, understood as the frequency difference between two adjacent frequency-domain samples, directly impacts the performance of the SE. Indeed, when the resolution is low, a narrowband signal component is represented by only a handful of frequency-domain samples, necessitating for the SE to operate with a small window length w_l (see Section 2.3.4 in Chapter 2 for a study on the impact of the window length parameter for support estimation). Working with a small window length typically reduces the performance of the SE because separating noise-only values from signal-plus-noise

values is more difficult with a shorter smoothing window. A solution to combat this phenomenon is to increase the resolution, which is done by raising the duration of signal acquisition. Indeed, the frequency resolution of n_{blocks} 20-µs-blocks is $\delta f = 1/(n_{blocks} \cdot 2 \cdot 10^{-5}) = 50/n_{blocks}$ kHz. Doing so allows the SE to operate with a higher window length and reduces the strength of spectral leakage. However, this comes at the expense of the time resolution. If a transmission's duration is shorter than the duration of acquisition, chances are that the transmission will not contribute enough energy to be detected successfully by the SE.

In a way, the choice of the window length w_l also has an impact of the trade-off between type I and type II errors. A too large window length would cause large side effects on the edges of the signal components, likely resulting in a higher P_{FA}, and a too little window length would decrease the performance of the SE in a low SNR regime, in the way that P_D and P_{FA} would be closer from one another, while we want them to be as close as possible to 1 and 0 respectively. Further, w_l needs to account for signal components with narrow bandwidths. For now, we set $w_l = 10n_{blocks}$ samples; this choice will be discussed in Section 4.4.4.2.

Note that the outcome of a support estimation carried out on n_{blocks} is a boolean $n_{blocks}M_i$-sized vector. To obtain a support estimate $\hat{S}_i$ of length M_i, we subdivide the $n_{blocks}M_i$ vector in M_i smaller vectors each of length n_{blocks}. A voting rule is applied to each of the M_i smaller vectors: if at least 20% of the n_{blocks} boolean elements are 1's, then the corresponding element of the M_i-sized support estimate is set to 1. This has not shown to reduce P_D and can even filter out sporadic type I errors, thus marginally reducing P_{FA}.

The impact of undersampling on the SE performance When the received signal is undersampled, the resolution of the signal components is not altered because sampling duration is unchanged. However, a lower number of samples is acquired, detrimentally impacting the performance of the SE. Because of spectrum folding due to undersampling, the occupancy ratio in the sample sequence goes up whenever the undersampling ratio decreases. As a consequence, even if the number of type I errors is unchanged, P_{FA} increases, simply because there is proportionally fewer samples that do not contain any signal component. In an extreme situation of a very low undersampling ratio, very few samples, if any, do not contain any useful information, resulting in an extremely high P_{FA} in the likely event of type I errors, if the P_{FA} is even computable in the first place (if every frequency-domain sample contains information, there is no possible type I error so P_{FA} cannot be computed).

4.4.4.2 Validation of the SE

We now present an experiment to quantify the performance of the SE. For each traffic scenario, the received signal is sampled, and the noise variance and partial support $\hat{S}_i$ are estimated. $\hat{S}_i$ is compared to the actual partial (folded) signal support S_i by computing P_D and P_{FA}. The simulation is averaged over 50 runs. The varying parameters are the following:

- the occupancy ratio ρ of the Nyquist-rate signal;
- the SNR;

- the duration of acquisition $\Delta_{acq} = n_{blocks} \cdot 2 \cdot 10^{-5}$;
- the undersampling ratio δ_i;
- the window length parameter w_l for the SE.

Two of these factors are not under our control, namely the SNR and the occupancy ratio at the Nyquist rate. Of these two factors, the SNR can be estimated after sampling, but before running the SE, by using the previously computed noise variance estimate using the following equation:

$$SNR = \frac{P_{signal}}{P_{noise}} \approx \frac{P_{signal+noise} - P_{noise}}{P_{noise}} = \frac{P_{received\ signal}}{P_{noise}} - 1, \tag{4.2}$$

where P denotes the power or energy and P_{noise} is the noise variance multiplied by the considered number of samples. Note that the approximation in (4.2) consists in the omission of cross terms in the computation of $P_{signal+noise}$.

The other factor, the occupancy ratio at the Nyquist rate, is unknown prior to SE, but can be found after SE is performed.

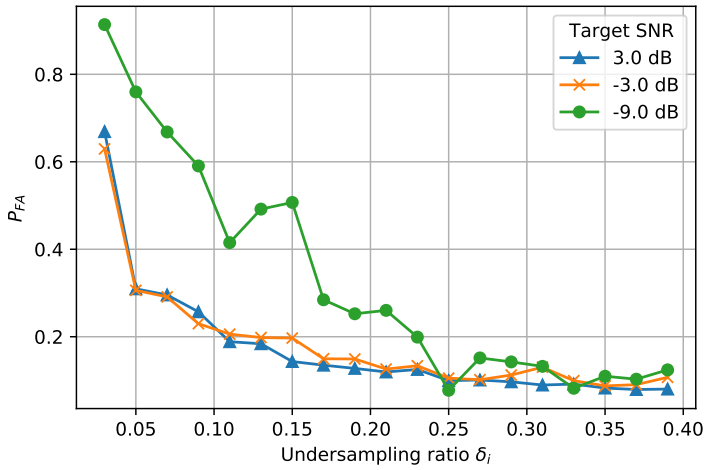
For each set of parameters, we filter out results for which $P_D < 0.99$ and consider P_{FA}.

Fig. 4.4 displays the P_{FA} corresponding to a value of $P_D \geq 0.99$ as a function of the undersampling ratio for the different traffic scenarios and for three SNR regimes. The number of 20µs blocks is $n_{blocks} = 20$, corresponding to a sampling duration of 400 µs and to a Nyquist-rate vector of size $Nn_{blocks} = 40000$ samples - but support estimation is performed on a much smaller vector, of size $Nn_{blocks}\delta_i$. The SE window length is set to $w_l = 10n_{blocks} = 200$ samples. A series of observations can be made from this experiment.

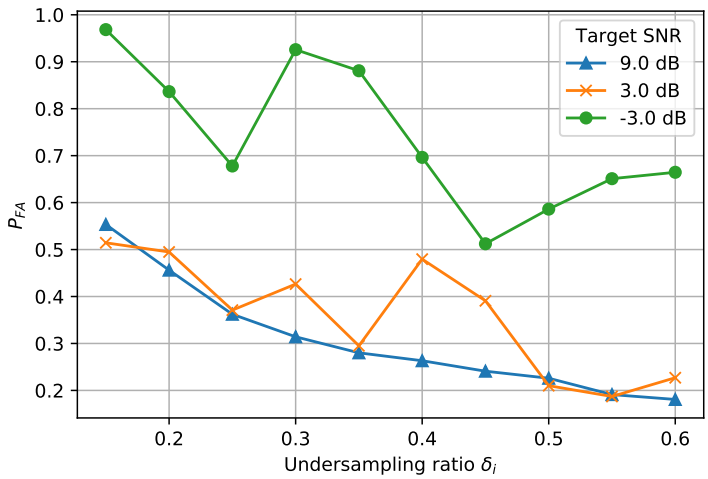
P_{FA} heavily depends on the traffic scenario and the undersampling ratio For a given traffic scenario, as the undersampling ratio goes up, P_{FA} declines. This is because an important contributor to type I errors is the side effect taking place at the edges of signal components. When the undersampling ratio increases, so does the number of noise-only samples, while the number of side-effect-related type I errors remains about the same, thus effectively diminishing P_{FA}.

For every traffic scenario, every SNR regime can be accomodated In all three considered SNR regimes, for every traffic scenario, it is possible to have $P_D \geq 0.99$ without $P_{FA} = 1$ (which indicates that the SE is of no use). However, depending on the undersampling ratio, the share of type I errors can be quite high. Generally speaking, the lower the SNR, the higher the P_{FA}; however, for every scenario, it seems like the high SNR and medium SNR regimes have about the same performance in terms of P_{FA}, meaning that we could have reached a P_{FA} lower-bound for this set of settings.

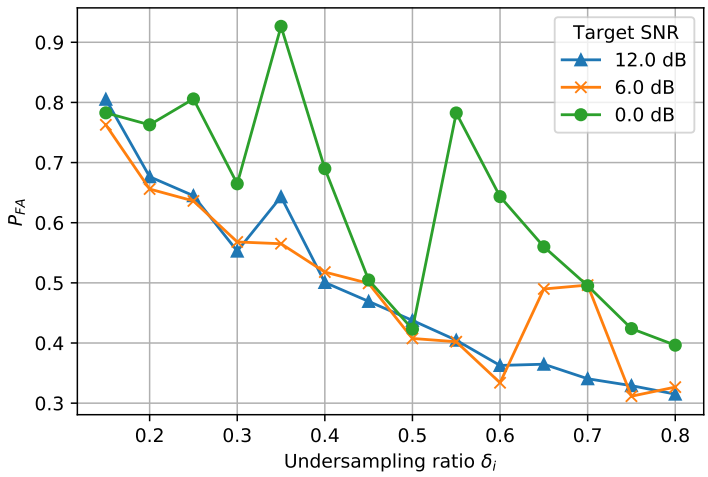
To gain more insight on the role of the other parameters, such as the acquisition duration represented by n_{blocks} or the SE window length w_l, we run some follow-up simulations, with the same metric (P_{FA} under $P_D \geq 0.99$) but with a variation of some parameters. We specifically focus on the high and medium SNR regimes as they offer possibilities to either reduce the



(a) Scenario 1: 2.5% average occupancy ratio.



(b) Scenario 2: 10% average occupancy ratio.



(c) Scenario 3: 25% average occupancy ratio.

Figure 4.4: Probability P_{FA} of type II errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i for various SNR regimes and traffic scenarios. The lower the P_{FA}, the better.

acquisition duration at no expense, or reduce P_{FA} through the window length parameter w_l of the SE.

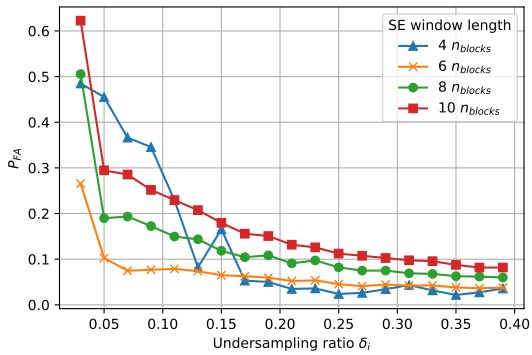
A higher SNR regime offers leverage to curb P_{FA} through the w_l setting A lot of type I errors are the expression of the side effect taking place at the edges of the different signal components in the frequency spectrum. The scope of this side effect heavily depends on the window length parameter w_l of the SE: the smaller w_l is, the lower the magnitude of the side effect. However, w_l is also useful because it smoothes noise-only values, making them easier to discriminate from signal-plus-noise values. Fig. 4.5 helps solve this trade-off for the high and medium SNR regimes. For this simulation, the window length varies from $w_l = 4n_{blocks}$ samples to $w_l = 10n_{blocks}$ samples.

Lowering the window length from $w_l = 10n_{blocks}$ to $w_l = 6n_{blocks}$ incurs massive gains in P_{FA}: consider Fig. 4.5(b), which represents the medium SNR regime in Scenario 1. For these settings, the P_{FA} for a 10%-of-Nyquist sampling rate drops from 24% for $w_l = 10n_{blocks}$ to a mere 7% for $w_l = 6n_{blocks}$. Similar gains can be obtained for various undersampling ratios, traffic scenarios and in the high and medium SNR regimes. Note that changing w_l does not seem to have a strong impact in the low SNR regime.

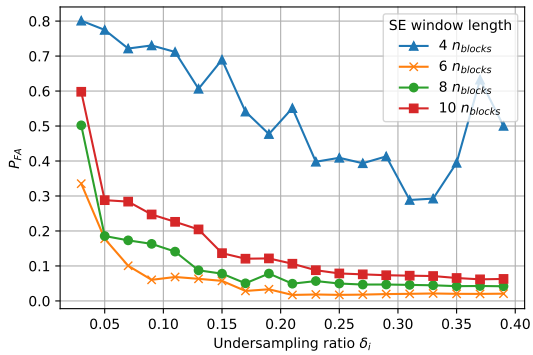
A downside of this parameter tuning is that reducing the window length too aggressively results in a spike in P_{FA}. To illustrate, consider $w_l = 4n_{blocks}$ samples in all panels of Fig. 4.5. It is not clear why P_{FA} increases to such an extent. It is perhaps due to the bandwidth of the narrowest signal components, which is precisely $4n_{blocks}$ in an Nn_{blocks} frequency-domain representation of the entire traffic sampled during $20n_{blocks}$ µs.

A higher SNR allows for lower acquisition durations Fig. 4.6 displays P_{FA} versus the undersampling ratio δ_i for different acquisition durations (represented by the number n_{blocks} of 20 µs blocks) in Scenario 1 for the high SNR regime (panel (a)) and the low SNR regime (panel (b)). The SE window length is $w_l = 10n_{blocks}$ samples. In the low SNR regime, reducing the number of 20-µs blocks (and thus the acquisition duration and number of points) results in a degradation of the P_{FA}. This means that the higher frequency resolution that comes at the cost of a lower time resolution is necessary in the lower SNR regime. The same logic applies to the intermediate SNR regime, not depicted for conciseness. However, in the high SNR regime, lowering the acquisition duration (and thus increasing the time resolution) can be done with no increase in P_{FA}. This opens the path towards an enhanced time resolution in high SNR regimes, which is useful for very brief signals. The same phenomenon appears in Scenario 2 and 3. Note that the combined effects of shortening the SE window length and the acquisition duration has not been tested; however, it is expected that both cannot be achieved at the same time (except perhaps for even higher SNRs), because one effect relies on a higher time resolution and the other on a higher frequency resolution, which are antagonistic.

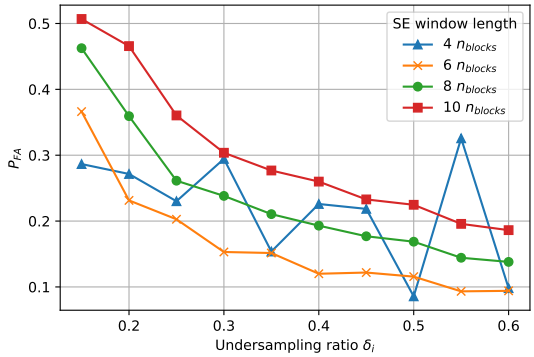
Impact of the noise variance estimation and thresholding step The noise variance estimation performed at the previous step using the KNVE is principally used to set the P_{FA}



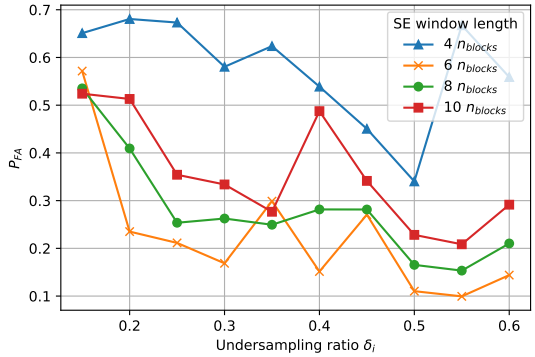
(a) Scenario 1, high SNR (SNR= 3 dB).



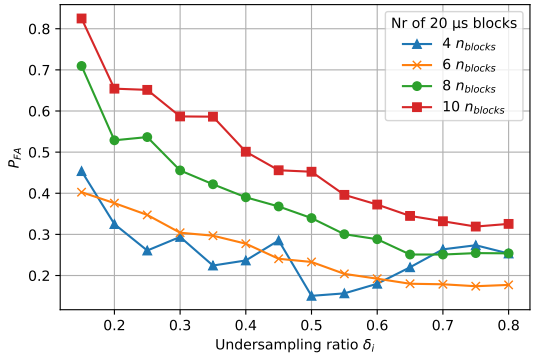
(b) Scenario 1, medium SNR (SNR= −3 dB).



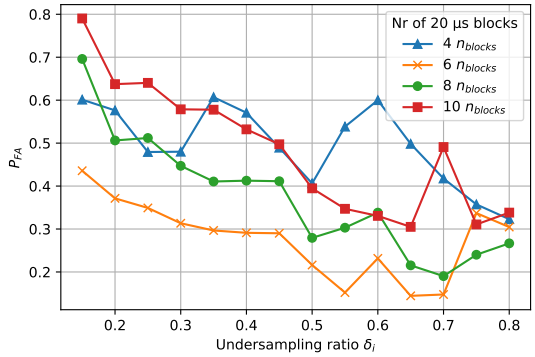
(c) Scenario 2, high SNR (SNR= 9 dB).



(d) Scenario 2, medium SNR (SNR= 3 dB).

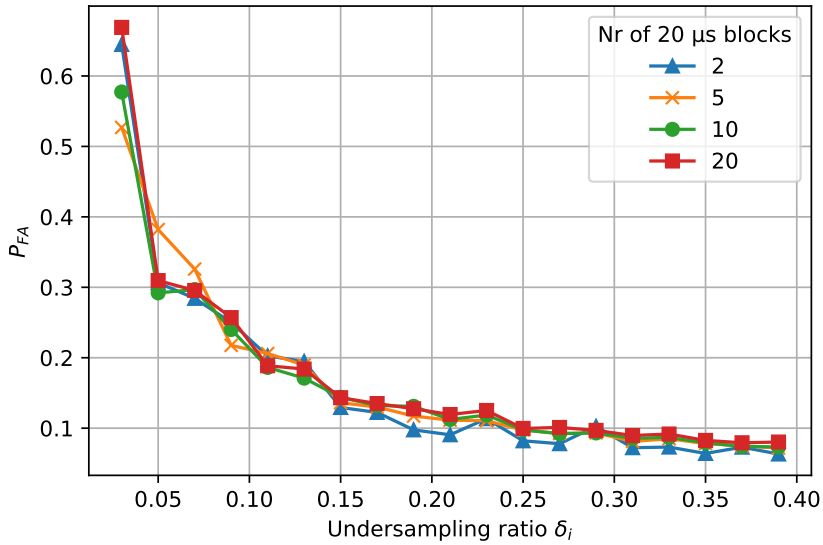


(e) Scenario 3, high SNR (SNR= 12 dB).

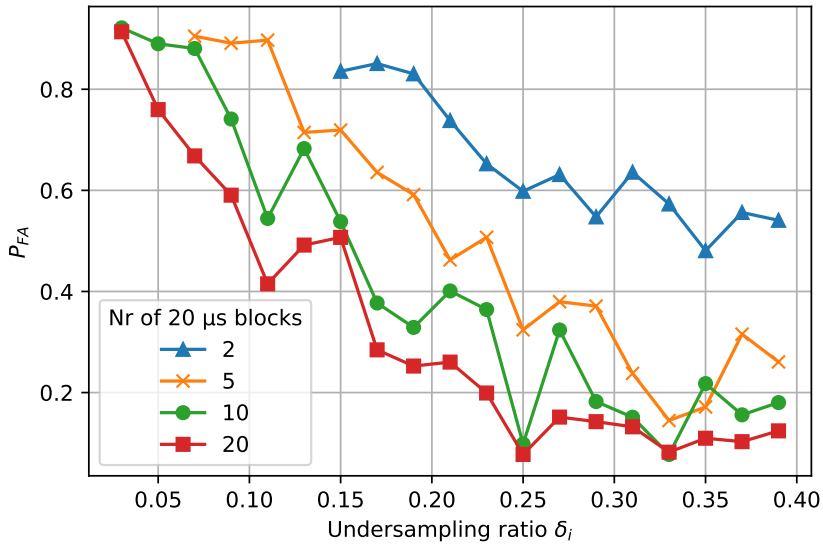


(f) Scenario 3, medium SNR (SNR= 6 dB).

Figure 4.5: Probability P_{FA} of type I errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i for various SE window lengths w_l, in different SNR regimes and traffic scenarios. The lower the P_{FA}, the better.



(a) Scenario 1 in the high SNR regime (SNR = 3 dB).



(b) Scenario 1 in the low SNR regime (SNR = −10 dB).

Figure 4.6: Probability P_{FA} of type I errors (under $P_D \geq 0.99$) versus undersampling ratio δ_i in traffic scenario 1 for various acquisition durations in the (a) high SNR regime; (b) low SNR regime. The lower the P_{FA}, the better.

threshold for the SE. However, even with a successful noise variance estimation, we have noticed that the measured P_{FA} vastly differs from the target P_{FA} used to set the threshold for detection. This discrepancy between the target P_{FA} and the measured P_{FA} is caused by an anomalous number of type II errors. We believe undersampling in each MRS branch to be the main source of type II errors not attributable to actual outliers of the noise distribution. The spectrum folding of the noise, signal components, and of their respective energies, probably calls for a revision of the equations of the SE, in a fashion similar to that of [140]. As a consequence, we had to set the P_{FA} threshold *a posteriori*, and we were not able to assess the degradation caused by an inaccurate noise variance estimation.

Beyond the problem of the discrepancy between target and measured P_{FA}, the support estimation step requires calibration. Indeed, Neyman-Pearson-theory-based energy detection seeks to maximize P_D for some fixed P_{FA} target. On the contrary, for our infrastructure, we wish to minimize P_{FA} for a fixed value of $P_D \geq 0.99$. This task is more difficult than the first one because through the KNVE, we primarily have information about noise (which is under $\mathcal{H}_0$), not so much about signal-of-interest components (which are under $\mathcal{H}_1$). Hence, a calibration step to set the threshold of the SE according to our objective (minimize P_{FA} for a fixed P_D) is an open issue and should be investigated.

For the remainder of this study, we will set this issue aside and use partial support estimates $\hat{S}_i$ for which the fulfillment of the detection condition ($P_D \geq 0.99$) is validated *a posteriori*.

4.4.5 Combination of partial support estimates

After partial support estimation has been carried out on the samples acquired at every MRS branch, a combination procedure is applied to obtain a new support estimate $\hat{S}$, representative of the received signal if it had been sampled at the Nyquist rate. This combination procedure was initially introduced in [20], and is presented in our context in Algorithm 4.1.

Algorithm 4.1: Support combination procedure of the MRS.

Input: Support estimates at each MRS branch $\{\hat{S}_i\}_{1 \leq i \leq L}$, block size N
Initialize: $\hat{S} = 0_N$
for $i = 1$; $i = i + 1$; $i \geq L$ **do**
  $\hat{S}_i^{exp} = \text{expand}(\hat{S}_i, N)$ (support expansion)
end
$\hat{S} = \prod_{1 \leq i \leq L} \hat{S}_i^{exp}$ (combination of expanded support)
Output: Full support estimate $\hat{S}$

In Algorithm 4.1, the expand function consists in considering the partial support estimate as a (frequency) period and reproducing this period until the length of the outcome reaches block size N (see Fig. 4.7 for an illustration of the expansion procedure).

The full support estimate $\hat{S}$ is crucial as it determines which Nyquist-rate samples will be reconstructed and which will be dropped. The study of the combination procedure will be separated in two parts. First, we are interested in its performance in optimal conditions, that

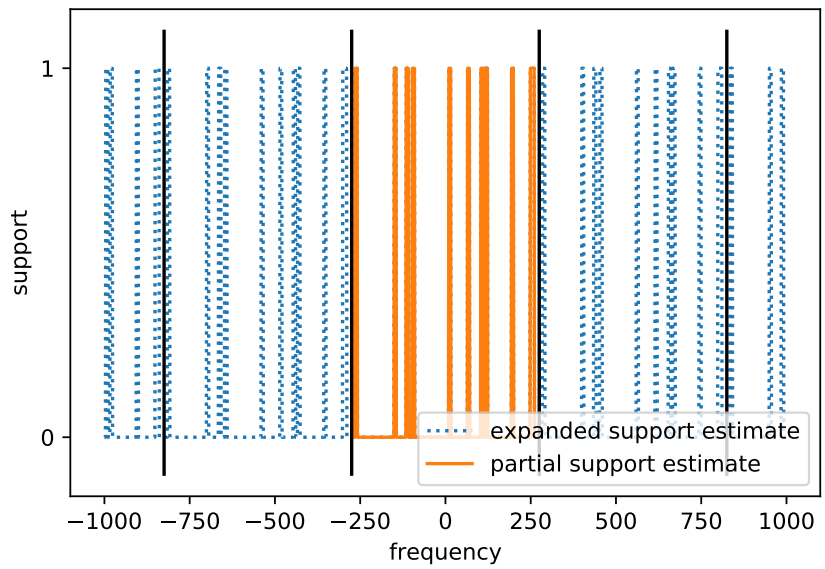


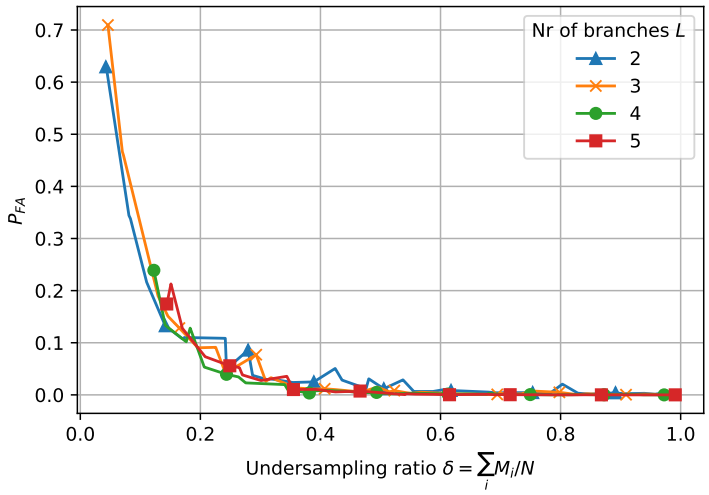
Figure 4.7: Illustration of the support expansion procedure. The estimates are centered around 0 for visual clarity. The M_i-sized partial support estimate (solid line) is periodically reproduced to form the N-sized expanded support estimate (dotted line).

is, its performance if the inputs are perfect partial support (a perfect partial support verifies $P_D = 1$ and $P_{FA} = 0$ and is obtained by multiplying the exact signal support S by the folding matrix F_i). Indeed, perfect $\{S_i\}$ do not necessarily result in an perfect full support estimate. The expansion procedure reproduces all aliases onto the total bandwidth, potentially entailing a lot of type I errors for the Nyquist-rate frequency-domain signal. Contrary to aliases, "original" signal components are at the same location in the frequency spectrum no matter what the sampling rate is, so the combination of the expanded support estimates should remove these type I errors. However, if there are not enough MRS branches and/or the sampling rates are too low, the combination procedure cannot solve the support so as to remove all type I errors. After studying the performance of the combination procedure in optimal conditions, we will test its resistance to faults. To do so, we will use partial support estimates $\{\hat{S}_i\}$ issued from the previous steps of our processing chain, with $P_D \geq 0.99$ and various values of P_{FA}.

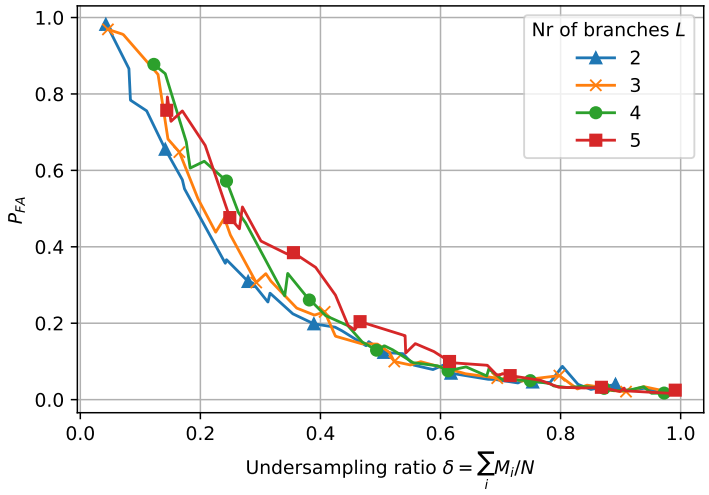
4.4.5.1 Performance of the combination procedure in optimal conditions

In this section, we use perfect partial supports $\{S_i\}$ (with $P_D = 1$ and $P_{FA} = 0$), and combine them to form a full support estimate $\hat{S}$. The varying parameters are the number L of MRS branches, the total undersampling ratio (defined as $\delta = \sum_i M_i/N$) and the traffic scenario. We then compare $\hat{S}$ to S and compute P_{FA} (P_D is always equal to 1 since every partial support estimate as a P_D of 1). The varying parameters are the traffic scenario, the number of branches L and the undersampling ratio δ. The simulation is averaged over 50 runs.

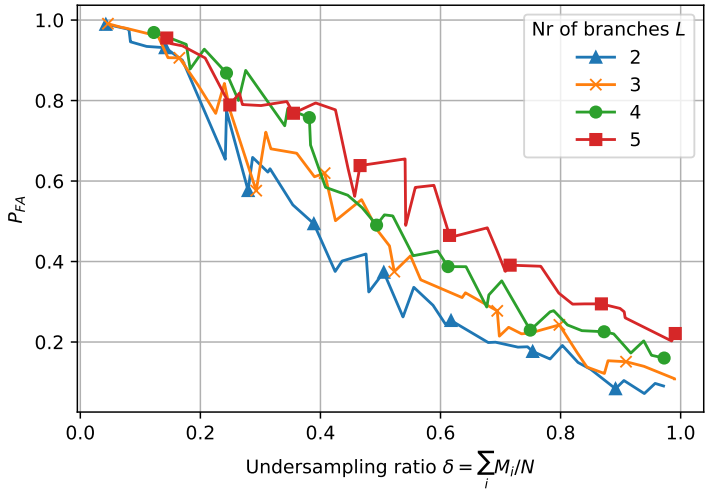
Fig. 4.8 displays the probability of false alarms P_{FA} as a function of the undersampling ratio $\delta = \sum_i M_i/N$ for different traffic scenarios and different number of branches L. We can see that to reach a low P_{FA}, higher occupancy ratios result in higher requirements for sampling



(a) Scenario 1: 2.5% average occupancy ratio.



(b) Scenario 2: 10% average occupancy ratio.



(c) Scenario 3: 25% average occupancy ratio.

Figure 4.8: P_{FA} versus undersampling ratio δ for support combination of L perfect partial supports ($P_D = 1$, $P_{FA} = 0$), each computed in an MRS branch.

rates.

For a 25% occupancy ratio (Scenario 3), it is generally not possible to perfectly estimate the signal support, even from perfect partial support estimates[5]. Yet, for $\sum_i M_i \approx N$, values of P_{FA} are rather low. This suggests that even for a high spectrum occupancy ratio, using the MRS can be beneficial: it allows to reduce the sampling rates in each branch (and thus relaxing constraints on the ADCs), although it does not permit a reduction of the number of acquired samples (compared to the Nyquist rate).

However, if the sampling rate in one branch is too low compared to the Nyquist-rate occupancy ratio, the samples collected in this branch are basically composed of aliases only, due to too much spectrum folding. The ability of this branch to contribute to support combination is therefore reduced. This is why in Scenario 3, the more branches an MRS has, the worse support combination turns out to be. Indeed, for a given undersampling ratio, the more branches, the smaller the sampling rate of each branch. For example, for $L = 5$ branches, the average sampling rate of a branch is 20%-of-Nyquist if $\delta = 1.0$. Compare this with the occupancy ratio of 25% in Scenario 3: it makes sense that the partial support in any branch of a $L = 5$-branch MRS is subject to heavy spectrum folding.

4.4.5.2 Combination of partial support estimates $\hat{S}_i$

In realistic operating conditions, we cannot use perfect partial supports $\{S_i\}$ but have to rely on partial support estimates $\{\hat{S}_i\}$. We now measure the P_{FA} versus undersampling ratio δ for different scenarios and SNR regimes. For traffic scenarios 1 and 2, the number of branches has a very limited impact on P_{FA}, so we restrict our study to MRS systems with $L = 4$ branches to simplify the analysis. For traffic scenario 3, the number of branches has a higher impact on P_{FA}. For conciseness, here we only present the results with $L = 4$ branches.

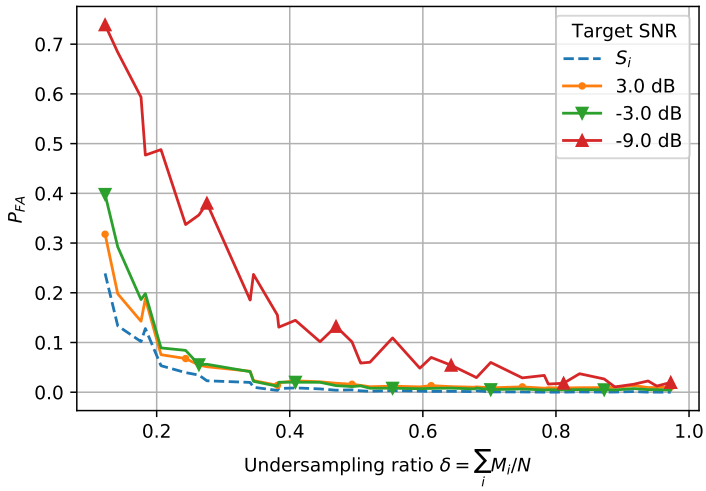
Fig. 4.9 represents support combination from support estimates $\{\hat{S}_i\}$ computed for various SNR regimes and for the three traffic scenarios. The dashed line represents the corresponding support combination carried out with perfect partial supports $\{S_i\}$ instead of support estimates: it corresponds to the optimal performance of the support combination step.

For every SNR regime, the P_{FA} follow the same trend as when support combination is carried out on perfect partial supports $\{S_i\}$ (dashed line), except that they are shifted up. Indeed, the type I errors in the partial support estimates $\{\hat{S}_i\}$ propagate into the support combination $\hat{S}$. The higher the SNR, the closer $\hat{S}$ is to the perfect support combination. An exception can be found in Scenario 3, for which results are better in the medium SNR regime than in the high SNR regime. We do not have an explanation to this phenomenon. Note that its significance may be low, as for Scenario 3, values of P_{FA} are very high even for high sampling rates anyway.

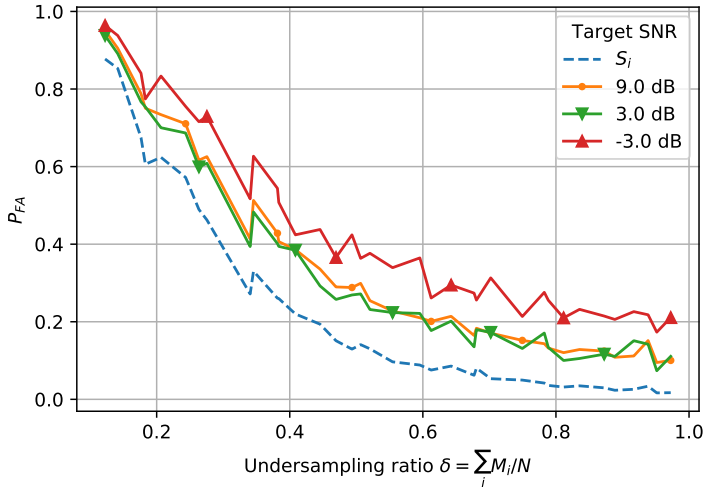
4.4.6 Conclusion

In this section, we have tried and validated our K-means-based noise variance estimator (KNVE), support estimator (SE). We also quantified the performance of the support combination proce-

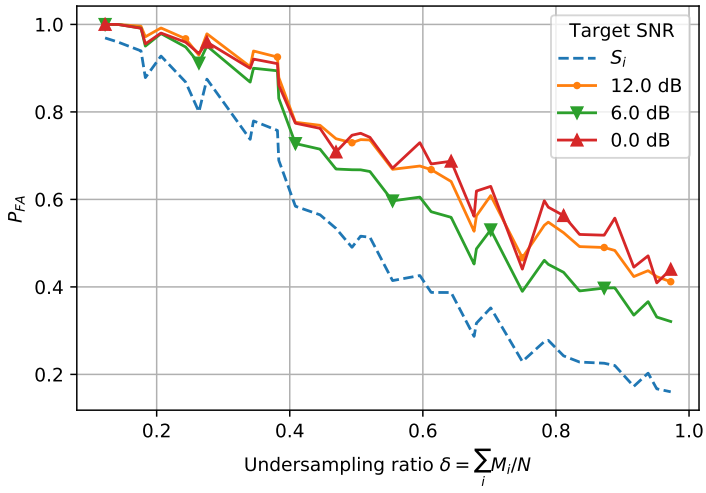
[5]This study is carried out with the following constraint: $\sum_i M_i \leq N$. If we relax this constraint, we can reach $P_{FA} = 0$ for the support combination.



(a) Scenario 1: 2.5% average occupancy ratio.



(b) Scenario 2: 10% average occupancy ratio.



(b) Scenario 3: 25% average occupancy ratio.

Figure 4.9: P_{FA} versus undersampling ratio δ for support combination of $L = 4$ partial support estimates, each computed in an MRS branch, for various SNR regimes. The dashed line represents the performance in optimal conditions.

dure.

Generally speaking, the low-traffic scenario allows for a significant reduction in total number of acquired samples $M = \sum_i M_i$ compared to sample acquisition at the Nyquist-rate. The medium-traffic scenario makes some reduction of M possible, while the high-traffic scenario does not. Nonetheless, even if the occupancy ratio is too high to permit a reduction in M, a relaxation of the sampling rate constraints on each ADC is achievable.

Now that we have described the conditions to obtain a satisfactory signal support estimate, we turn to the reconstruction of the MRS-acquired signal, a critical component of our proposed sampling infrastructure.

4.5 Impact of the reduction procedure and solver

4.5.1 Introduction

After determining the frequency-domain support of the signal of interest $\hat{S}$, the next step in our processing chain is the recovery, or reconstruction, of Nyquist-rate samples $\hat{Y}$ from the sub-Nyquist MRS samples $\{Z_i\}_i$. It is conducted whenever there is a request from the client to recover the messages transmitted by their deployed devices. Signal recovery consists in solving the CS USLE and is carried out on-demand: as it requires a substantial amount of computational resources, it would be pointless to perform it if no one benefits from it.

Several strategies towards signal recovery can be implemented. Some rely on support estimation while some do not. Generally speaking, the reduction procedure permitted by the support estimation step allows for faster and more accurate USLE[6] resolution. The gain in speed is provided by the fact that the reduced USLE is smaller than the non-reduced one, while the gain in accuracy is a consequence of the reduction of the USLE solution space. However, these gains can be lowered and even counterbalanced by a defective support estimation. For every solving method relying on the support estimation and reduction procedure, we compare the outcome of signal recovery carried out with our support estimate $\hat{S}$ to that with the actual support S. Indeed, not only is this a good way to see how a solver copes with an imperfect support estimation, but it can also depict a realistic scenario where a client actually knows the locations in time and frequency of their devices' transmissions.

To measure the performance of the different solvers, we look at the difference between received frequency-domain samples Y hypothetically sampled at the Nyquist rate and reconstructed samples $\hat{Y}$. This difference is only evaluated in the support S of the Nyquist-rate samples. This is done in an effort to restrict the evaluation to that of our sampling infrastructure and not include impairments beyond the scope of our study, such as Rayleigh fading. We use the relative root-mean-square error (relative RMSE) metric:

$$RMSE_{rel} = \frac{||\hat{Y}[S] - Y[S]||_2}{||Y[S]||_2}. \tag{4.3}$$

[6]Strictly speaking, the "Underdetermined" part (U) of the denomination "USLE" might not even be true once the reduction procedure is applied, in which case there is oftentimes more equations than unknowns. Nonetheless, for simplicity, we still (abusively) refer to the system to solve as an USLE.

This metric has its flaws. In particular, the relationship between the relative RMSE and an information retrieval metric such as the BER is *a priori* not linear. Indeed, reconstructed samples can be as suitable as Nyquist-rate samples for retrieval of transmitted messages, yet have a substantial relative RMSE, simply because due to noise folding, the AWGN[7] in the reconstructed samples has different realizations than in the Nyquist-rate samples. Nonetheless, it remains a useful indicator to estimate the degradation caused by our sampling infrastructure and make qualitative comparisons between solving methods and MRS settings.

We also consider the time performance of the different solvers. Although signal recovery is a somewhat complex step, some solving methods are simpler than others. The conditions of simulations are such that the resolution times are not necessarily representative of a real-life implementation designed for this specific purpose, however these resolution times can be the support of a discussion and comparison of the complexities of the different solvers.

In the reminder of this section, we will first introduce the different tested solvers, then turn to the analysis of their performance based on simulations carried out in different setups. Finally, we will draw insights for choosing the solver that is the most adapted to a given situation.

4.5.2 Presentation of tested solvers

To solve the CS USLE, we have tested 8 different solvers. Two are based on convex optimization (ℓ_1-minimization), three on greedy optimization, two on ℓ_2-minimization, and one specific to the MRS based on alias resolution. The other families of solvers (combinatorial methods, Bayesian methods, and so on) presented or mentioned in Chapter 1 are not covered because we were not able to identify a solver which was applicable to the MRS. Neither did we explore the variants of greedy algorithms and of convex optimization methods.

4.5.2.1 Solvers based on ℓ_1-minimization: CVXOPT and CVXOPT-RP

For the USLE resolution based on ℓ_1-minimization, we use the CVXPY [188] package. It is a general-purpose convex optimization program to solve constrained Basis Pursuit Denoising (BPDN) (1.29). When applied to the non-reduced USLE, we call the solver CVXOPT (short for ConVeX OPTimization). When applied to the reduced USLE after the reduction procedure, it is referred to as CVXOPT-RP (short for CVXOPT-Reduction Procedure).

Note that in the constraint of (1.29), there is an error parameter called ϵ which is not trivial to set. If set too low, the problem is infeasible (as per CVXPY terminology); if set too high, the solution is suboptimal. To overcome this issue, we implemented a simple heuristic that consisted in trying to solve the problem for a low value of ϵ, and if the problem turned out to be infeasible, we progressively increased ϵ until the problem was feasible or after an iteration counter was exceeded. This potentially resulted in execution times higher than what CVXOPT and CVXOPT-RP are capable of.

[7] Throughout this section, we only consider noisy signals coming from the previous steps of our sampling infrastructure, and leave noiseless signals aside.

4.5.2.2 Solvers based on ℓ_2-minimization: PINV and PINV-RP

The solvers based on ℓ_2-minimization find the least-squares solution to the USLE. The ℓ_2 norm tends to promote energy over sparsity, which is promoted by the ℓ_0-pseudonorm and, to a lesser extent, by the ℓ_1-norm. Therefore, it seems that ℓ_2-minimization would be less efficient than other approaches: we will see that this depends on the considered settings. These solvers are based on the computation of the Moore-Penrose pseudo-inverse (MPPI) $A^\dagger$ of the sampling matrix A (or of its reduced version A_{red}). After the computation of the MPPI, it is multiplied by the observation vector $Z = [Z_1 \dots Z_L]^T$ to yield an estimate $\hat{Y}$ of Y. The names of the two solvers are drawn from numpy.linalg.pinv, the name of the Numpy method for the computation of the MPPI. The solvers come in two flavors: PINV (without the reduction procedure) and PINV-RP (for which the reduction procedure is applied).

PINV is extremely fast: after the MPPI of A is computed once and for all, applying PINV to every block consists in a mere matrix multiplication $A^\dagger Z$. In contrast, every time the support estimate changes, so does the reduced sampling matrix A_{red}, and PINV-RP has to compute a new MPPI at each evolution of the support estimate. However, these evolutions can be somewhat sporadic (at least compared to the duration of one block) and most importantly, the reduced sampling matrix A_{red} can be much smaller than A, entailing considerable gains in execution times. A downside of PINV-RP is that it can be numerically unstable, especially when the support estimate is bad.

4.5.2.3 Solvers based on greedy optimization: OMP-SE, OMP-CV and OMP-RP

For the study of the behavior of greedy algorithms on our MRS-sampled signals, we use an implementation of Orthogonal Matching Pursuit (OMP) [189] present in the Scikit-learn package for Python. We compare three flavors of OMP. As we recall from Chapter 1, OMP needs a stopping criterion, generally in the form of the degree of sparsity k. OMP-SE (short for OMP-Support Estimation) operates on the non-reduced USLE, but relies on the support estimation previously carried out to determine the number of frequency-domain samples in the support of the signal of interest (proxy for the number of non-zeros in a noisy setting) and use it as its stopping criterion. OMP-CV does not take the support estimate as an input but uses cross-validation to estimate the degree of sparsity and fix its stopping criterion. Finally, OMP-RP is OMP-SE applied to the reduced USLE.

Because an internal step of OMP is based on ℓ_2-minimization, OMP-RP shares similitudes with PINV-RP.

4.5.2.4 Solver based on alias resolution: AMRS

Finally, a solver specific to the MRS and presented in the context of the AMRS [113] was tested. This solver does not rely on any equation resolution. The principle of this solver is, for each element in the signal support estimate, to find an MRS branch in which the corresponding element is not aliased with any other signal component, and to select this corresponding element. If there is no branch in which the corresponding element is non-aliased, then it is considered

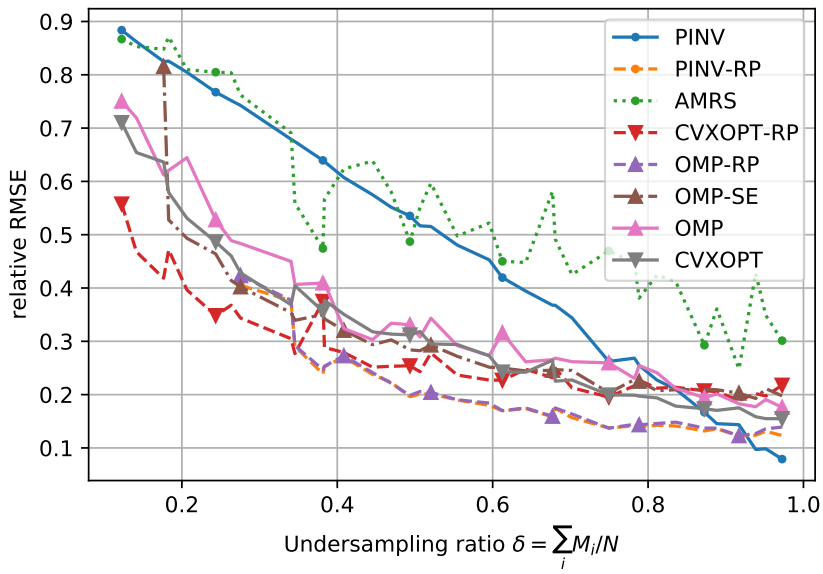


Figure 4.10: Relative RMSE versus undersampling ratio δ for traffic scenario 1 (low traffic) in the high SNR regime (SNR = 3 dB).

non-retrievable and arbitrarily set to zero in the vector solution. By essence, this solver relies on support estimation and cannot be applied without the support estimation step.

Now, we turn to the analysis of simulation results.

4.5.3 Simulation results

We compare the performance of the different solvers for different traffic scenarios, SNR regimes and undersampling ratios δ. For each setting, we have conducted support estimation on $n_{blocks} = 20$ blocks and with a smoothing window of length $w_l = 6n_{blocks} = 120$ samples. Each solver was tested (when applicable) with our support estimate and with the actual signal support. Each simulation point is averaged over 5 runs.

Additionally, we tested every solver for $L \in \{2, 3, 4, 5\}$. However, all other settings being equal, no significant variation in solver performance has been observed when L was modified. Therefore, for the remainder of this section, we will present results obtained with an $L = 4$-branch MRS.

4.5.3.1 Traffic scenario 1 (low traffic), high SNR regime

This is the most favorable set of settings. Fig. 4.10 depicts the performance of the different solvers.

Solver performance in terms of relative RMSE Apart from AMRS and PINV, all the solvers have more or less the same performance, though PINV-RP and OMP-RP are a little better than the others for higher undersampling ratios. Although what can be considered a satisfactory relative RMSE depends on a variety of factors, notably the choice of the signal

Solver	Execution time (s)	Ratio To Fastest
PINV	.013	fastest
PINV-RP	.021	1.6
OMP-RP	.039	3.1
AMRS	.047	3.7
CVXOPT-RP	.104	8.2
OMP-SE	.272	21
CVXOPT	1.70	134
OMP	2.29	180

Table 4.3: Execution times (in seconds) for the different solvers for traffic scenario 1 (low traffic) in the high SNR regime (SNR = 3 dB).

modulation and the implemented error correction protocol, we can see that a relative RMSE of 0.2 is reached for $\delta = .5$ with OMP-RP and PINV-RP, meaning that 4 branches can sample at an average rate of 12.5% of Nyquist and recover the sampled signal fairly well.

PINV's performance increases linearly with the undersampling ratio. The regularity with which the relative RMSE decreases when δ increases is quite an interesting phenomenon, even more so as we will see that it takes place for all the other sets of settings.

AMRS has a lower performance than all other solvers, a trend that we will witness in every other set of settings.

Execution time Though the performance in terms of relative RMSE is about the same for most solvers, the same cannot be said about execution times. Again, the elapsed times are to be taken cautiously as they were run on a setup that is not dedicated to this task. Yet, comparing these execution times helps us understand which solver could be adequate and which could not. Except from the AMRS, which was not thoroughly optimized (however, the relative RMSE performance *a priori* excludes it anyway), all solvers rely on third-party packages that have been optimized by their respective authors. Table 4.3 displays these execution times in the ascending order. All times are in seconds. Values are averaged over the number of branches L and the undersampling ratio δ.

PINV is the fastest solver. Following are the reduction-procedure-based solvers, which take advantage of the signal sparsity and therefore of the smaller problem size. In this pack there is also AMRS, which does not rely on equation solving. Solvers not based on the reduction procedure have to cope with a much larger problem and are thus slower. Cross-validation and hyperparameter selection (which concerns OMP and CVXOPT) take their toll in the form of a much longer execution time.

4.5.3.2 Traffic scenario 2 (medium traffic), high SNR regime

While still in a high SNR regime, we now increase the traffic. Fig. 4.11 depicts the performance of the different solvers.

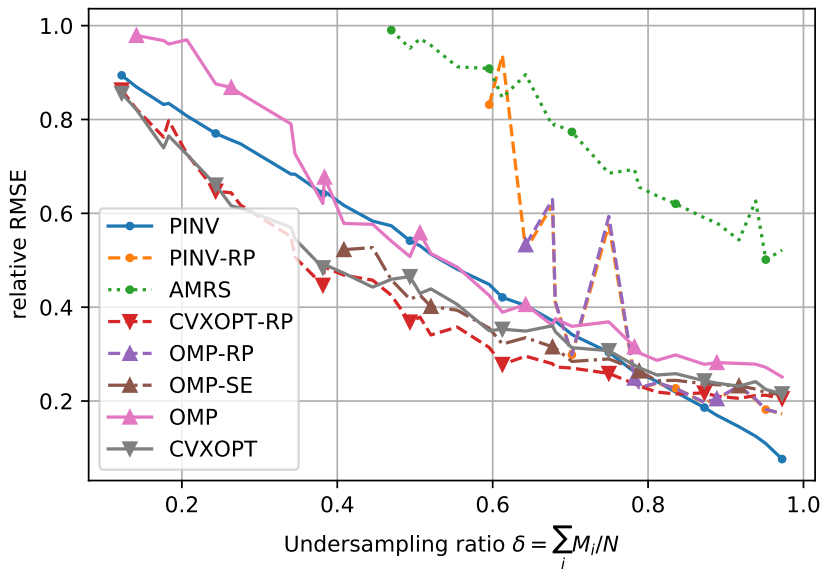


Figure 4.11: Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB).

Solver performance in terms of relative RMSE For high undersampling ratio, all solvers (except AMRS and PINV) have about the same performance. However, for lower values of δ, solvers based on support estimation (with the notable exception of CVXOPT-RP) output relative RMSEs above 1.0 and/or numerically unstable results. This outlines the crucial role of correct support estimation. As a basis of comparison, consider Fig. 4.12 (CVXOPT and OMP are not depicted because their performance are the same as in Fig. 4.11). It is clear that when the actual signal support is known, performance of reduction-procedure-based solvers is much better than when the signal support is estimated (especially for low sampling rates).

Execution time Table 4.4 displays execution times in the ascending order. All times are in seconds. Values are averaged over the number of branches L and the undersampling ratio δ.

All reduction-procedure-based solvers take a severe hit in terms of execution times. However, this is mostly due to the bad support estimation for low sampling rates. Indeed, the execution times for these solvers with the actual signal support, displayed in Table 4.5, show only a twofold increase, instead of the more-than-tenfold increase witnessed when solvers are applied on support estimates.

4.5.3.3 Traffic scenario 3 (high traffic), high SNR regime

In the most demanding traffic scenario, the phenomena at stake with the medium traffic scenario are still present and even exacerbated. Fig. 4.13 depicts the performance of the different solvers.

Solver performance in terms of relative RMSE The solvers based on support estimation do not converge if the undersampling ratio is not very high (of at least $\delta \geq 0.85$ for PINV-RP

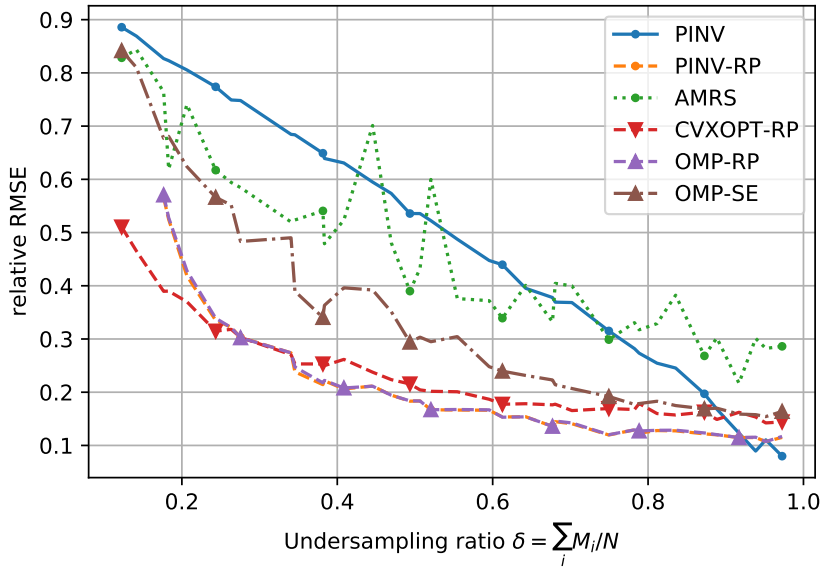


Figure 4.12: Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). Solvers use the actual signal support for USLE resolution (when applicable).

Solver	Execution time (s)	Ratio To Fastest
PINV	.013	fastest
AMRS	.059	4.7
PINV-RP	.330	26
CVXOPT-RP	.507	40
OMP-RP	.671	53
CVXOPT	1.62	126
OMP-SE	2.02	158
OMP	2.40	188

Table 4.4: Execution times for the different solvers for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB).

Solver	Execution time (s)	Ratio To Fastest
PINV	.013	fastest
PINV-RP	.057	4.2
OMP-RP	.102	7.6
CVXOPT-RP	.104	7.7
OMP-SE	.509	38

Table 4.5: Execution times for the different solvers for traffic scenario 2 (medium traffic) in the high SNR regime (SNR = 9 dB). Solvers use the actual signal support for USLE resolution (when applicable).

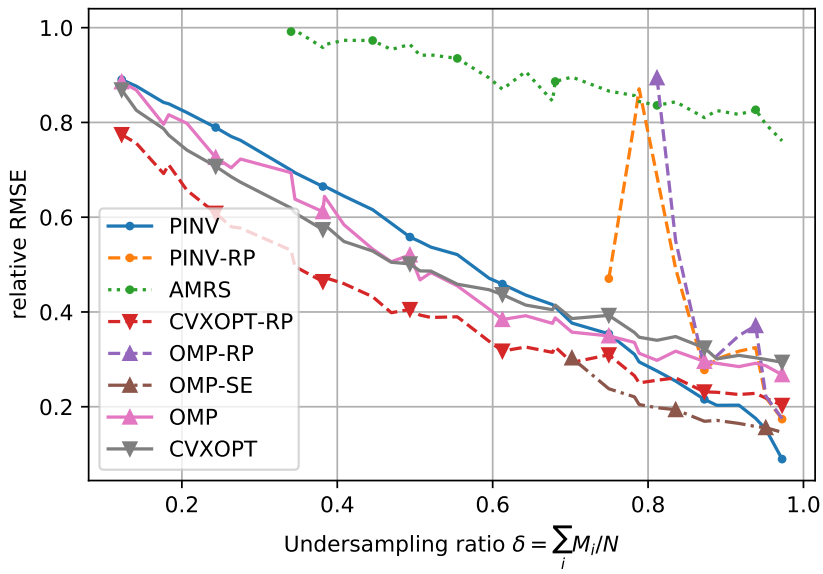


Figure 4.13: Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB).

and OMP-RP, $\delta \sim 0.70$ for OMP-SE). Again, CVXOPT-RP is the notable exception and is even the best solver for lower undersampling ratios. The performance of PINV-RP and OMP-RP increases when the actual signal support is used instead of the support estimate, as can be seen in Fig. 4.14.

Execution time Table 4.6 displays execution times in the ascending order. All times are in seconds. Values are averaged over the number of branches L and the undersampling ratio δ.

The reduction-procedure-based solvers are once again taking a toll in terms of execution time. Noticeably, CVXOPT-RP, OMP-RP and OMP-SE now take longer than their non-reduction-procedure-based counterparts. For OMP-RP and OMP-SE, this could be because the stopping criterion, based on the support estimate, entails a higher number of iterations

Solver	Execution time (s)	Ratio To Fastest
PINV	.013	fastest
AMRS	.058	4.7
PINV-RP	1.07	85
CVXOPT	1.76	141
OMP	2.35	188
CVXOPT-RP	2.50	200
OMP-RP	4.52	361
OMP-SE	6.80	543

Table 4.6: Execution times for the different solvers for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB).

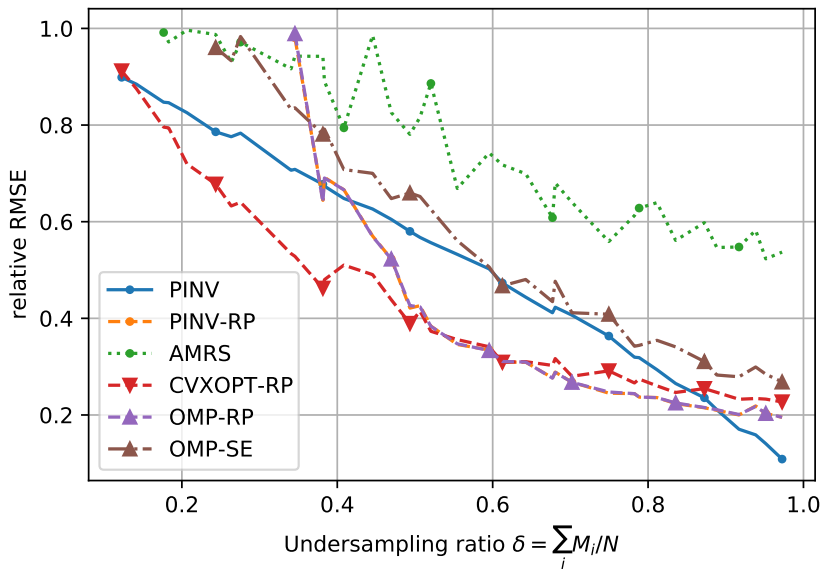


Figure 4.14: Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). Solvers use the actual signal support for USLE resolution (when applicable).

Solver	Execution time (s)	Ratio to fastest
PINV	.013	fastest
CVXOPT-RP	.238	19
PINV-RP	.306	24
OMP-RP	.604	47
OMP-SE	1.71	134

Table 4.7: Execution times for the different solvers for traffic scenario 3 (high traffic) in the high SNR regime (SNR = 12 dB). Solvers use the actual signal support for USLE resolution (when applicable).

than OMP (due to a pollution of the support estimate by type I errors in lower undersampling ratios). For CVXOPT-RP, this can be because the reduced problem is infeasible (due to type I error pollution) except with a high ϵ parameter: our construction of CVXOPT-RP could therefore be the source of the longer execution time.

Similarly to the previous settings (medium traffic, high SNR regime), the reduction-procedure-based solvers become faster when the actual support is used instead of the support estimate, as seen in Table 4.7.

4.5.3.4 Traffic scenario 1 (low traffic), medium SNR regime

Fig. 4.15 depicts simulation results for an MRS in traffic scenario 1 in a medium SNR regime.

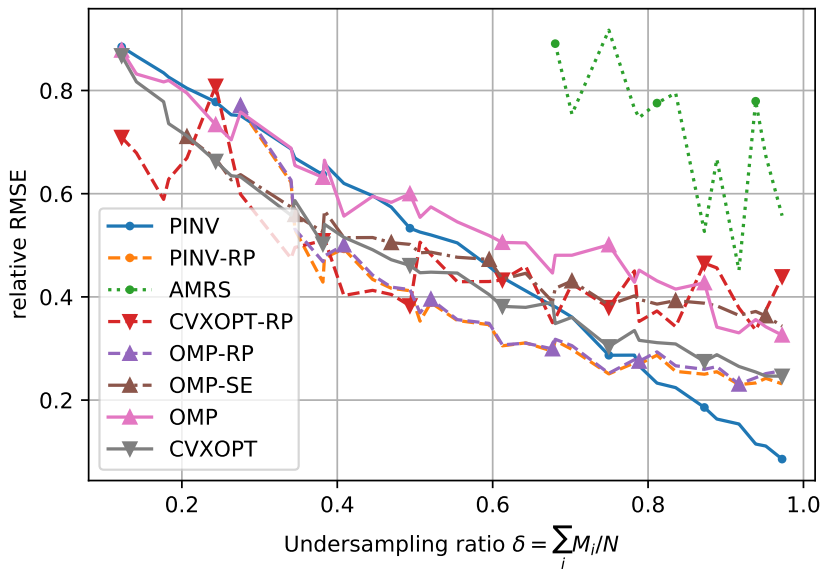


Figure 4.15: Relative RMSE versus undersampling ratio δ for traffic scenario 1 (low traffic) in the medium SNR regime (SNR $= -3$ dB).

Solver performance in terms of relative RMSE For traffic scenario 1, in the medium SNR regime, the solvers behave exactly in the same way than in the high SNR regime, except the relative RMSE curves are shifted upwards.

Execution time As depicted in Table 4.8, the execution times are extremely similar to those in the high SNR regime. Support estimation is very good for many undersampling ratios, leading to a low P_{FA} for support estimation: reduction-procedure-based solvers are not polluted by an excessive amount of type I errors.

Solver	Execution time (s)	Ratio to fastest
PINV	.013	fastest
PINV-RP	.020	1.5
OMP-RP	.035	2.7
AMRS	.046	3.7
CVXOPT-RP	.118	9.4
OMP-SE	.252	20
CVXOPT	1.56	124
OMP	2.26	179

Table 4.8: Execution times for the different solvers for traffic scenario 1 (low traffic) in the medium SNR regime (SNR $= -3$ dB).

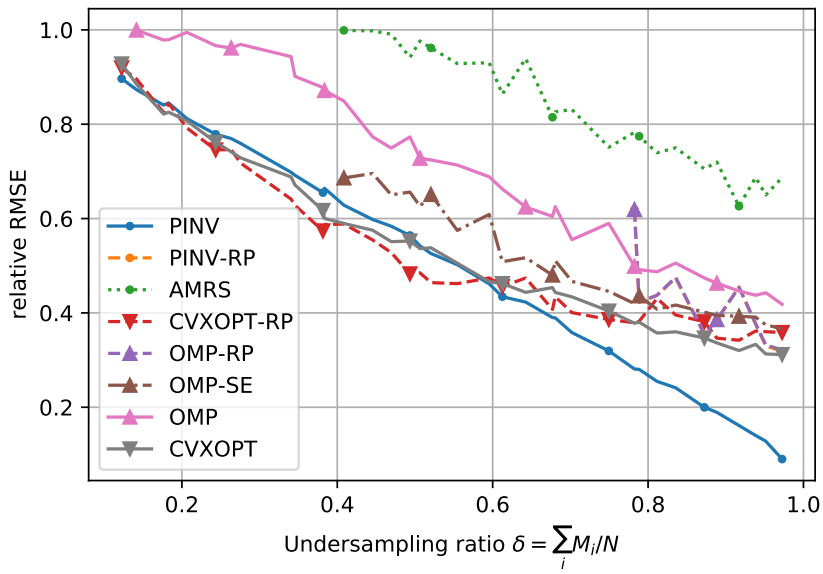


Figure 4.16: Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the medium SNR regime (SNR = 3 dB).

4.5.3.5 Traffic scenario 2 (medium traffic), medium SNR regime

Fig. 4.16 depicts simulation results for an MRS in traffic scenario 2 in a medium SNR regime.

Solver performance in terms of relative RMSE In the medium traffic scenario and medium SNR regime, there starts to be a significant degradation of the performance of all solvers except PINV, to the point that PINV outperforms all the other solvers, except CVXOPT and CVXOPT-RP for undersampling ratios under $\delta \sim 0.5$. Of course, this is in part due to the inexact support estimation, as reduction-procedure-based solvers beat PINV when applied with the actual signal support as input, as shown in Fig. 4.17.

Execution time The execution times for scenario 2 in the medium SNR regime are very similar to that in the high SNR regime. No matter whether the support estimate or the actual support is used for the support-estimate-based solvers, this is true in both cases.

4.5.3.6 Traffic scenario 3 (high traffic), medium SNR regime

Fig. 4.18 depicts simulation results for an MRS in traffic scenario 3 in a medium SNR regime.

Solver performance in terms of relative RMSE In the high traffic scenario and medium SNR regime, PINV has almost the same performance as CVXOPT and CVXOPT-RP, except for higher sampling rates ($\delta \geq 0.75$) where PINV outperforms CVXOPT and CVXOPT-RP. Other solvers result in a higher relative RMSE, whatever the undersampling ratio. With these settings, even using the actual signal support as input does not improve the performance of support-estimation-based solvers, as shown in Fig. 4.19.

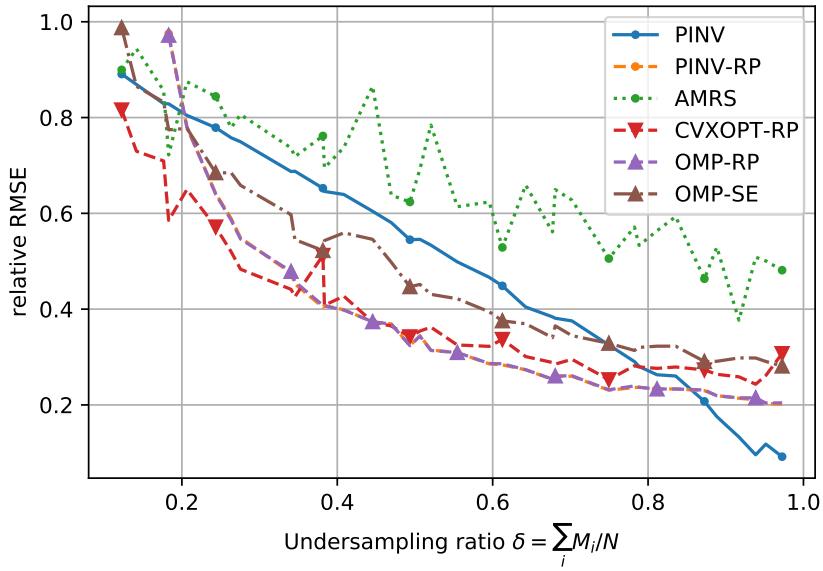


Figure 4.17: Relative RMSE versus undersampling ratio δ for traffic scenario 2 (medium traffic) in the medium SNR regime (SNR = 3 dB). Solvers use the actual signal support for USLE resolution (when applicable).

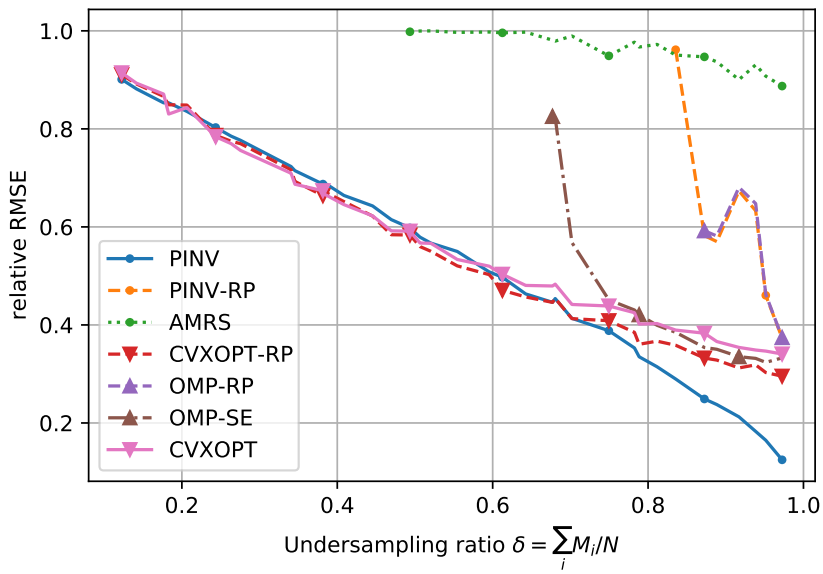


Figure 4.18: Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the medium SNR regime (SNR = 6 dB).

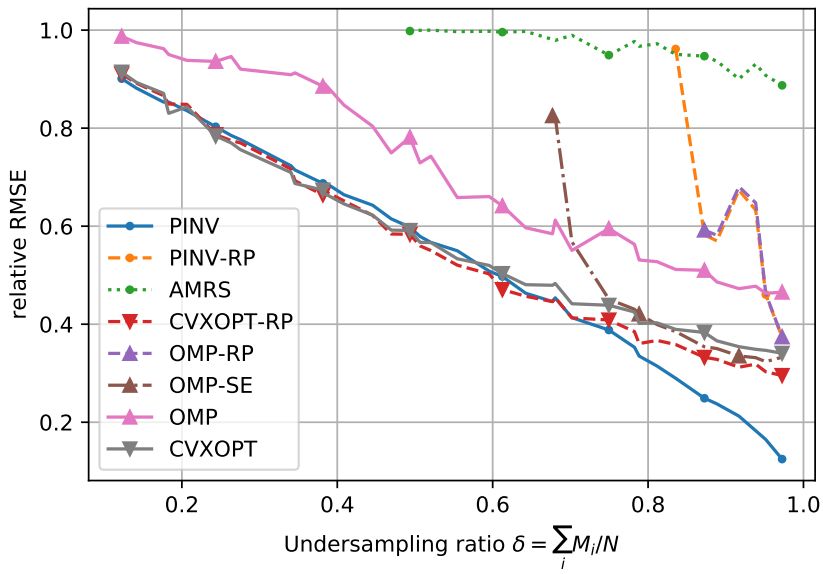


Figure 4.19: Relative RMSE versus undersampling ratio δ for traffic scenario 3 (high traffic) in the medium SNR regime (SNR = 3 dB). Solvers use the actual signal support for USLE resolution (when applicable).

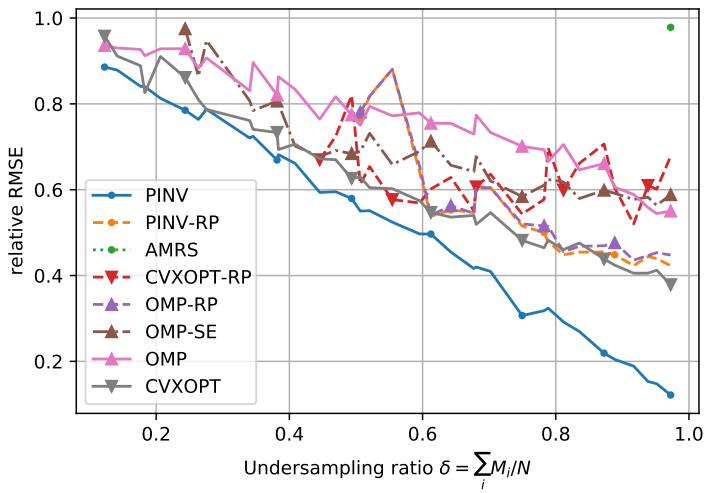
Execution time The execution times for scenario 3 in the medium SNR regime are very similar to that in the high SNR regime. No matter whether the support estimate or the actual support is used for the support-estimate-based solvers, this is true in both cases.

4.5.4 Low SNR regime

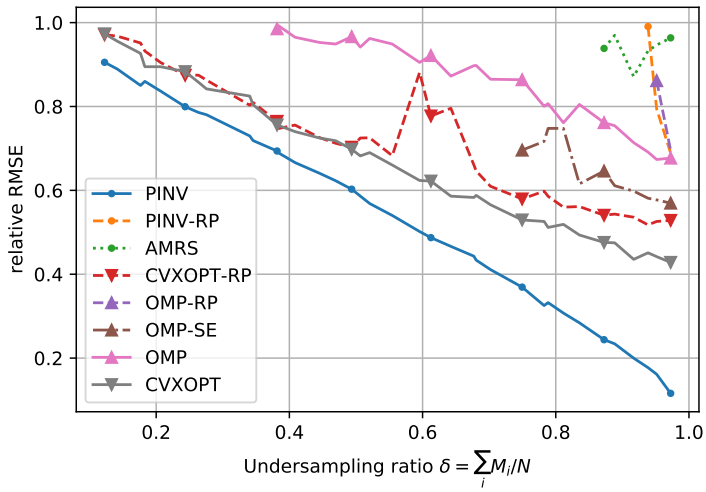
In the low SNR regime, it does not matter whether the support estimate or the actual support is used as input to reduction-procedure-based solvers. The traffic scenario does not matter either, nor does the undersampling ratio. Whatever the settings, in the low SNR regime, PINV is always the best solver. Fig. 4.20 displays the results for all traffic scenarios in the low SNR regime, with reconstruction carried out with signal support estimates $\hat{S}$.

Solver performance in terms of relative RMSE PINV outperforms every other solver for every traffic scenario and every undersampling ratio, except in a few sporadic, insignificant occasions.

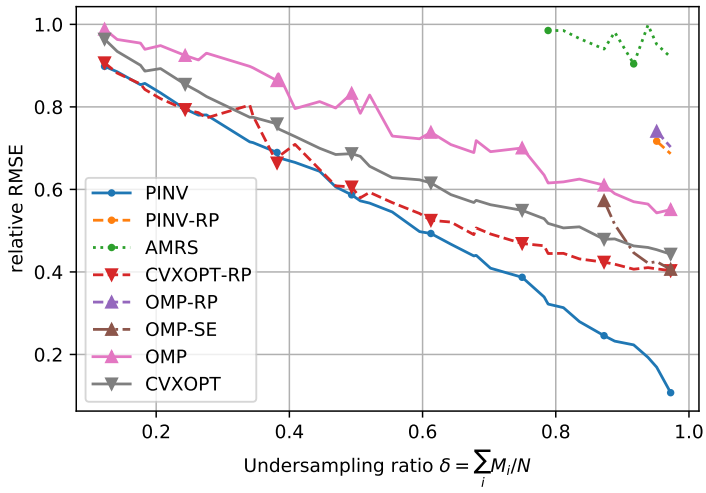
Note that these results must be taken with a pinch of salt. It is possible that the other solvers assign values to the solution $\hat{X}$ that result in a high relative RMSE, just because noise folding affects reconstruction with these solvers in a different manner than PINV, in a manner does not necessarily prevent information retrieval. However, only extracting the information from the samples and computing a metric such as the BER will tell us whether or not this is the case.



(a) Scenario 1: 2.5% average occupancy ratio.



(b) Scenario 2: 10% average occupancy ratio.



(b) Scenario 3: 25% average occupancy ratio.

Figure 4.20: Relative RMSE versus undersampling ratio δ for all three traffic scenarios in the low SNR regime (SNR = -9 dB for Scenario 1, -3 dB for Scenario 2 and 0 dB for Scenario 3).

Execution time PINV is several orders of magnitude faster than all the other solvers (except AMRS).

4.5.5 Discussion on the performance of solvers

The first conclusion to be drawn is that no solver is optimal in every situation[8]. Therefore, for different conditions, it is recommended to pick the solver that works best under said conditions, and the results presented in this section can help with this task.

Generally speaking, the reduction procedure of the MRS plays an important part in the success of signal recovery. When the SE operates under favorable conditions (high SNR regime or medium SNR regime with low or medium traffic), the reduction procedure of the MRS provides a decisive improvement to the quality of signal recovery. Otherwise, the reduction procedure is of little use and sometimes even detrimental to signal recovery: it is then better to use a solver not based on the reduction procedure. However, it is not clear why greedy optimization and ℓ_1 minimization algorithms not based on the reduction procedure perform worse than PINV in unfavorable conditions (low SNR regime or medium SNR regime with high traffic).

In any case, the main conditions which should drive the choice of the solver are mainly the SNR regime and the traffic scenario. The choice of the best solver also depends on the undersampling ratio: for high undersampling ratios ($\delta \approx 1$), PINV is always better than the other solvers, but there are strong contenders for every other undersampling ratio. An exception is in the low SNR regime where PINV is better than every other solver no matter what the undersampling ratio is.

Along with the performance in terms of relative RMSE, one should also consider the execution times when choosing a solver, especially if the signal recovery procedure is intended to frequently take place. PINV is always the fastest, but reduction-procedure-based solvers have execution times of the same order of magnitude as PINV when traffic and SNR conditions are clement. Such favorable conditions are the same that allow for a high performance of reduction-procedure-based solvers. Meanwhile, non-reduction-based solvers are orders of magnitude slower than reduction-procedure-based solvers in favorable traffic and SNR conditions and PINV.

Taking into account performance in terms of relative RMSE and execution times, Table 4.9 helps pick the best solver depending on the traffic scenario and SNR regime. In this table, we do not consider the fraction of very high undersampling ratios ($\delta \approx 1$) where PINV performs best.

4.5.6 Conclusion on the analysis of solvers

In this section, we have studied the resolution of MRS-originated USLE systems by different solvers. Some of these solvers depend on the reduction procedure allowed by the support estimation step carried out earlier, while some do not.

[8]However, it seems that PINV and CVXOPT-RP take the lion's share in terms of applicability.

[9]CVXOPT-RP is to be preferred if time is not a strong constraint, PINV should be favored otherwise.

	Traffic scenario		
	1 (low traffic)	2 (medium traffic)	3 (high traffic)
High SNR	if $\delta \leq .35$: CVXOPT-RP PINV-RP/OMP-RP otherwise	CVXOPT-RP	$\delta \leq .7$: CVXOPT-RP OMP-SE otherwise
Medium SNR	if $\delta \leq .5$: CVXOPT-RP PINV-RP/OMP-RP otherwise	CVXOPT-RP/PINV[9]	PINV
Low SNR	PINV	PINV	PINV

Table 4.9: Best solvers for different traffic scenarios and SNR regimes.

Performance of the different solvers depend mainly on the traffic scenario, the SNR regime and the undersampling ratio. There is no almighty solver that outperforms all other solvers in every situation. Instead, we have identified the solver that fits each set of conditions the most.

Signal reconstruction is the ultimate step of our sampling infrastructure. Now, for the infrastructure to capitalize on its cognitive capabilities, we explore possible ways towards infrastructure adaptivity to changes in the radio environment and discuss implementable heuristics.

4.6 Towards infrastructure adaptivity

We have studied the performance of our sampling infrastructure in various environments with a diversity of parameters. Depending on the context, some infrastructure parameters are more sensible than others. For our infrastructure to be adaptive, it is therefore crucial to identify the environmental characteristics very quickly, if not instantly, and to be able to perform a live adjustment of the infrastructure parameters.

In this section, we first identify the radio environment characteristics impacting the performance of the MRS-based sampling infrastructure, and how they can be quantified in near real-time. Then, we introduce the relevant levers to adapt the infrastructure to changes in the radio environment.

4.6.1 Characteristics impacting the performance of the MRS-based infrastructure

We have identified two main exogenous factors that alter the performance of the infrastructure: the noise level and the spectrum occupancy ratio. The first factor is modeled in our study by high, medium and low SNR regimes, and the second by three traffic scenarios. The way these factors affect the performance of our sampling infrastructure has been detailed throughout this chapter; consequently, we do not cover it again here. We now explain how to estimate the SNR regime and spectrum occupancy ratio. An advantage is that this can be done in (relatively) early stages of the processing chain of the proposed infrastructure and notably before the on-demand step of signal reconstruction, giving the possibility to use the most adequate solver to whichever radio context *after* the acquisition of samples.

Determination of the SNR regime After the noise variance is estimated, as presented in Section 4.4.3, the SNR can be computed in each branch[10] using (4.2). However, the SNR regime does not solely depend on the SNR, but also on the spectrum occupancy (see next paragraph on ascertaining the spectrum occupancy ratio).

Ascertaining the spectrum occupancy ratio The support combination step (see Section 4.4.5) allows to compute the signal support estimate $\hat{S}$, which can then be used to find the spectrum occupancy ratio. The spectrum occupancy ratio is defined by $\rho = K/N$, where K is the number of non-zeros in S, the support of Y. As a consequence, an estimate of the spectrum occupancy ratio is $\hat{\rho} = \hat{K}/N$, where $\hat{K}$ is the number of non-zeros in $\hat{S}$.

4.6.2 Adapting to radio environmental changes

Now that we have identified and explained how to quantify the exogenous factors impacting the performance of the sampling infrastructure, we turn to the presentation of the control levers that can be activated and what should drive the decision to activate them. On a system level, these levers are endogenous factors that impact the sampling infrastructure performance.

Adapting the sampling rates Tuning the sampling rates is a powerful way to modify the performance of the sampling infrastructure. This can be done either by switching on (resp. off) an MRS branch to increase (resp. decrease) the number of measurements M. However, the number of active branches should never be lower than 2. This can also be done by changing the sampling rate(s) of one (or more) ADC(s) of the MRS, if we assume the clocks driving the ADCs to be reconfigurable. Note that the number of branches is, generally speaking, not impactful.

The main criterion that should drive a change in sampling rates is an evolution of the spectrum occupancy ratio. If this ratio goes up, the number of measurements M should go up, either by activating an inactivated MRS branch or by increasing the sampling rate(s) of one or several ADC(s). If the spectrum occupancy ratio is reduced, the number of measurements should go down - except in the event of a low SNR, in which case it is recommended that the number of measurements M be close to N.

Fig. 4.21 depicts a decision tree to help tune the sampling rates, starting from a number of measurements $M \approx N$. This heuristic can be run at each change in radio environment. A rule of thumb is that the number of measurements M and thus the sampling rates can be reduced (from Nyquist) if the SNR is not low and if the spectrum occupancy ratio is not high. Otherwise, our infrastructure can still be used, with the advantage that the sampling rate requirements of the ADCs are reduced compared to having one ADC sampling at the Nyquist rate.

[10]The SNR should be the same in every branch, up to an error factor. If there is a discrepancy between the different branches, this can be an indication that noise variance estimation has been unsuccessful in a branch. Comparing the SNRs at each branch, or even the noise variance estimates, is a good way to provide robustness to KNVE failure.

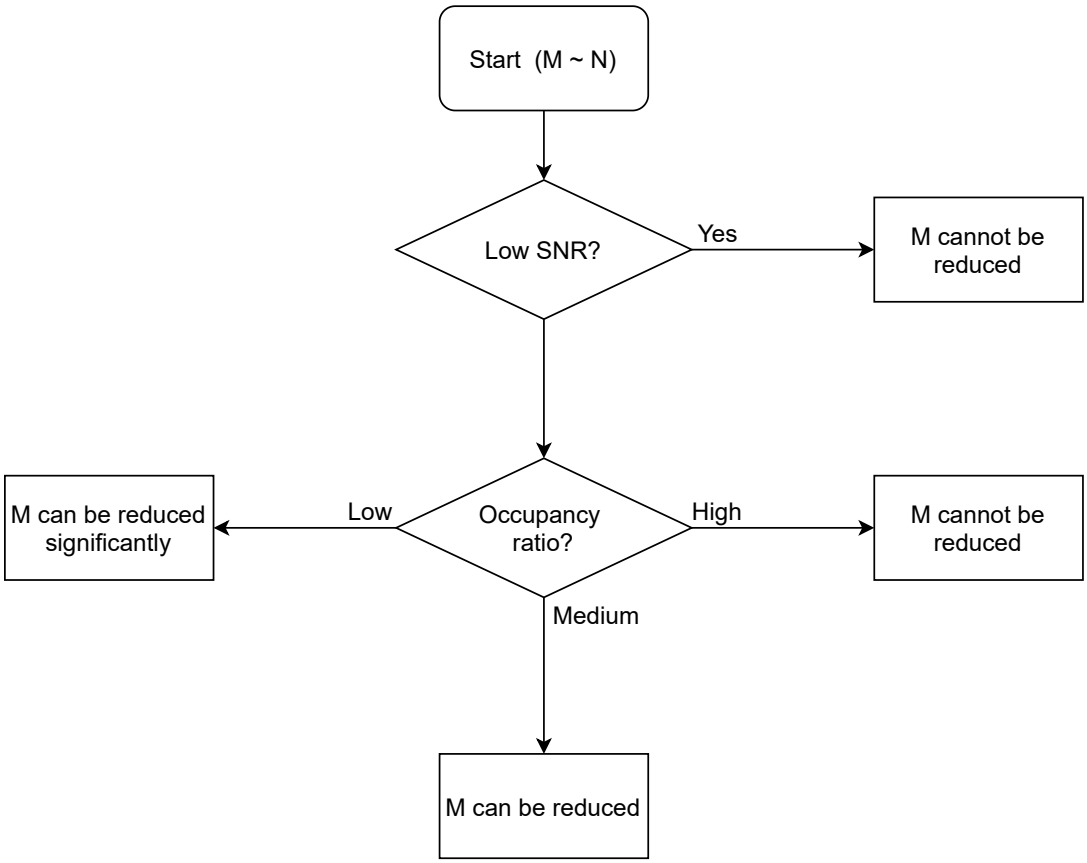


Figure 4.21: Decision tree to help tune the number of measurements M (and thus the sampling rates M_i).

Switching solvers As seen in Section 4.5, there is no one-size-fits-all, optimal solver. On the other hand, for a given radio context, there are solvers that have shown to perform better than the others, as summarized in Table 4.9. Nonetheless, though the choice of the solver has an impact on recovery performance, it cannot compensate for the loss of information incurred by sampling rates too low for the radio context.

4.6.3 Conclusion on infrastructure adaptivity

Our proposed sampling infrastructure exhibits several elements that make it prone to adaptivity. First, quantifiable exogenous factors impacting infrastructure performance have been identified. Second, levers to tune relevant parameters in near real-time have been explored.

4.7 Summary of our experimental findings

This chapter was dedicated to the presentation and analysis of our MRS-based, IoT-oriented sampling infrastructure prototype. We now summarize the content of this chapter.

First, we introduced our sampling infrastructure prototype and functionally described its different components. These components include an analog front-end for signal reception and preprocessing, an MRS composed of L parallel branches, blocks for the estimation of characteristics of the received signal (namely the KNVE, SE and SC blocks), a database for sample storage and a solver.

We then presented our IoT-emulating traffic generator. This simulatory traffic generator relies on reasonable assumptions to provide signals to the sampling infrastructure. The traffic generator is configured to output signals, based on IoT transmissions, that correspond to various scenarios of SNR regimes and traffic loads. A constraint was to have a moderate number of varying parameters (in order to keep the evaluation of the sampling infrastructure tractable) while being able to analyze the performance of the infrastructure in diverse realistic conditions.

Afterwards, we proceeded to evaluate four key blocks of the sampling infrastructure: the KNVE, the SE, the SC and the signal recovery block. The KNVE and SE have shown to be robust to moderate infringements of the formalized hypotheses of signal compacity and sparsity. For signal recovery, we have analyzed the performance of several solvers on signals with diverse characteristics of SNR and traffic load. We have found that there is no optimal solver that outperforms the other solvers in every situation; however, for specific radio environment conditions and MRS settings, it is possible to pick an optimal solver. For these four key blocks of the sampling infrastructure, a rule of thumb is that overall reduction of the sampling rate can be carried out under favorable radio conditions (low or medium traffic load, high or medium SNR). In less favorable conditions (high traffic load, low SNR), the overall sampling rate cannot be much reduced from the Nyquist rate without adverse effects on signal reconstruction. In this case however, since the MRS operates with parallel ADCs, each ADC can sample at a rate quite below that of Nyquist.

Depending on the radio environment conditions, the parameters of the sampling infrastructure can be modified to provide satisfactory gains in sampling rates while complying with

signal reconstruction requirements. In this regard, an adaptive sampling infrastructure can be extremely valuable. Such an infrastructure can detect changes in the radio environment and automatically change its settings accordingly. In this chapter, we discussed some steps towards infrastructure adaptability and how they could be implemented in our sampling infrastructure.

In the general conclusion of this document, we will explore future perspectives to improve upon our proof-of-concept mono-site sampling infrastructure prototype. In particular, we will discuss the extension towards a multi-site infrastructure and detail its stakes and remaining challenges.

Conclusion and perspectives

Summary of our contributions and perspectives

The problem at the heart of this PhD thesis work was the efficient sampling of sparse telecommunications signals. The sparsity in telecommunications can be seen as a form of sporadicity of communications in some domain *e.g.* signals that are sporadic in time, in frequency or in space. We addressed the problem of efficient sampling within the Compressed Sensing (CS) framework, which allows, under certain conditions, to lower the amount of acquired samples in comparison with continuously sampling at the Nyquist rate at multiple sites. We have mainly considered signal sparsity in the frequency domain. Nonetheless, sparsity in the time domain was also taken into account: indeed, time-domain sparsity consists in the absence of information-bearing signal components in the received signal in given time intervals. In this case, the frequency-domain spectrum during such intervals is effectively free of signal components, and the sampling parameters can be set accordingly.

An enabler towards reducing the number of acquired samples is the knowledge of the radio environment, in line with the Cognitive Radio (CR) paradigm. Acquiring knowledge of the radio context can be performed with Spectrum Sensing (SS) techniques. It allows to locate the radio communications in time, frequency and space, thus providing crucial information to tune the sampling parameters.

Our contribution in Chapter 2 of this document consisted in several cognitive tools, that is, tools that allow a receiver to sense its radio environment. The first tool is a K-means-based noise variance estimator (KNVE), capable to blindly estimate the variance of an AWGN channel with sparse signals. The second tool is a signal support estimator (SE), which estimates the frequency-domain support of a frequency-sparse received signal. Both the KNVE and the SE have a low complexity and have two moderate requirements of sparsity and compacity of the received signal. Both estimators rely on preliminary signal windowing with a rectangular window of size w_l samples. The choice of the w_l parameter has a substantial impact on KNVE and SE performance in terms of detection probability: it would be interesting to understand the underlying phenomena involved when windowing is carried out, in an effort to set w_l correctly. Another open question is whether choosing another averaging window would lead to a better performance of the SE and KNVE.

Sampling a signal at the Nyquist rate is fairly simple and can be carried out using a low-pass filter and an ADC operating uniformly. On the contrary, sub-Nyquist sampling requires a more complex setup. In this thesis, we focused on the multi-rate sampler (MRS), a CS sampling

scheme based on several parallel branches uniformly sampling the same signal at different rates, each of them lower than the Nyquist rate. Chapter 3 aimed at correctly setting the parameters of the MRS scheme. First, we have shown that if we represent the MRS sampling rates as integer multiples of a time quantum, a very important condition for an MRS to thrive is that the different sampling rates be coprime integers. We have also linked the number of samples to the quality of recovery of the original signal. Generally, an increase in number of measurements entails a better signal recovery. However, there are local variations in performance, meaning that some sets of sampling rates seem to consistently result in a higher recovery performance than others (at a comparable number of measurements). It would be interesting to identify these "special" sets, comprehend what makes them better and especially to characterize whether they entail higher performance with *any* sparse signal. Finally, we tried to quantify the impact of having different number of parallel branches, but the results we obtained were difficult to generalize. It seems that in most cases, the number of branches has a limited impact on system performance, but a more extensive study on this parameter would help settle the question.

In Chapter 4, we proposed an MRS-based sampling infrastructure targeted at sampling messages transmitted by Internet of Things (IoT) devices.

To provide a framework for the evaluation of our proposed sampling infrastructure, we introduced an IoT traffic generator, whose purpose was to simulate a realistic radio environment. To keep the number of varying settings and simulation results tractable, we made some simplifying assumptions about this traffic generator, that we now review. First, the reception power of all transmissions is the same. The assumption of an equally-shared transmission power throughout all users is reasonable, especially if a power control scheme is implemented; however, it is unlikely that transmissions from different IoT devices would arrive at the reception antenna with the same power. More realistically, each transmission would probably arrive with a different in-band SNR. This can be viewed as a manifestation of sparsity due to the geometric characteristics of signal propagation, and paves the way for a signal-component-by-signal-component recovery procedure. Second, we implemented a time-varying, yet flat channel. This assumption would likely be contradicted by reality, even more so as we consider a wideband channel. Mitigating this discrepancy involves implementing a more complex channel, with frequency-selective fades, requiring an additional step of channel equalizing. Third, we simulated different modulation standards using the same waveform (filtered QPSK). We envisioned a technology-agnostic sampling platform, and though it is impossible to predict what the IoT modulations of tomorrow will be, it would be interesting to contemplate the sampling infrastructure performance on signals with different waveforms.

Our proposed infrastructure samples the received signals at different sub-Nyquist rates in parallel branches, then relies on the building blocks introduced in Chapter 2 to compute partial signal frequency-domain support estimations (estimations of the signal support as seen in samples acquired in each branch). This step is based on the Neyman-Pearson theory, in which we seek to maximize the probability of detection for a given probability of false alarms. However, for our purpose, we would like to minimize the probability of false alarms for a given probability of detection. A decisive improvement would be to formalize a way to do so, ideally with the

least possible amount of information about the received signal of interest. After estimating the signal support in each MRS branch, a combination procedure is applied. This yields a support estimate of the signal as if it were sampled at the Nyquist rate.

The Nyquist-rate samples can now be recovered. This is done by solving an underdetermined system of linear equations (USLE) related to the settings of the MRS. We provided a comprehensive benchmarking of different USLE solvers (the method or algorithm used to solve an USLE) under diverse traffic loads and SNR regimes. The solvers we have selected are representative of some families of solvers, including those based on greedy optimization, ℓ_1-minimization and ℓ_2-minimization. In particular, the performance of an ℓ_1-minimization based solver named CVXOPT is promising, and it would likely be enhanced if the setting of the error parameter ϵ of CVXOPT was investigated more thoroughly.

Each family of solvers is composed of various solvers, each with its specific steps and characteristics. Generally speaking, we only benchmarked one method per family. As a consequence, for each family, it is possible that other solvers would perform better than the one tested, so marginal gains could be obtained through a more comprehensive benchmark in each family of solvers. It would also be interesting to benchmark solvers based on other paradigms, such as those based on Bayesian inference.

A conclusion of our benchmark is that there is no one-size-fits-all solver that outperforms the other solvers in every set of radio environment conditions; however, depending on the SNR regime and traffic load, some solver is to be preferred to the others. In Table 4.9, we described which solver to choose depending on the conditions.

We have seen in Chapter 4 that under low traffic loads (signal that are sparse in the time and frequency domains), the number of acquired samples can be significantly reduced, while under high traffic loads, it is impossible to successfully retrieve the information contained in signals if the number of acquired samples is not of the same magnitude of the hypothetical number of samples acquired through Nyquist rate sampling. As a consequence, a very important aspect of our sampling infrastructure is its capability to adapt to the changes in radio environment conditions. We have identified relevant environmental conditions impacting the performance of the sampling infrastructure, and the levers that can be activated to adapt to these conditions. A difficulty is to make sure that these levers (mainly the solver and the sampling rates of the sampling infrastructure) are indeed adapted to the environment conditions.

A future endeavor could include the development of heuristics to tune the relevant infrastructure parameters. In particular, we believe that artificial intelligence, and especially neural networks, have the potential to satisfactorily conduct this task. To use a neural network, we need a differentiable parametrable program, whose parameters can be updated through the gradient descent of a differentiable error function. We believe it is possible to propose such a program and error function for this problem of infrastructure adaptation, although their specifics remain to be defined. To benefit from accurate predictions from a neural network, we also need to have a large training dataset, but such data is likely to be fairly abundant.

In this thesis, we have mainly considered sparsity in the time and frequency domains. On the contrary, we have not mentioned sparsity in the space domain, and more generally speaking, the

phenomena at stake when spatial diversity comes into play. We now explore some opportunities and challenges of a multi-site sampling infrastructure, as envisioned in the general introduction of the manuscript. A particular attention to these should be paid by anyone who wants to extend our proposed mono-site sampling infrastructure to a multi-site one.

Extension: towards a multi-site sampling infrastructure

The main difference between a mono-site infrastructure and a multi-site one is that in the latter, the reception antennas are geographically distinct from one another. Fig. 5.1 reproduces the envisioned multi-site sampling infrastructure described in the general introduction of this document. We suppose that each reception site is equipped with a reception antenna and one or several MRS branch(es), and that the outcoming samples are then sent to a fusion center for further processing.

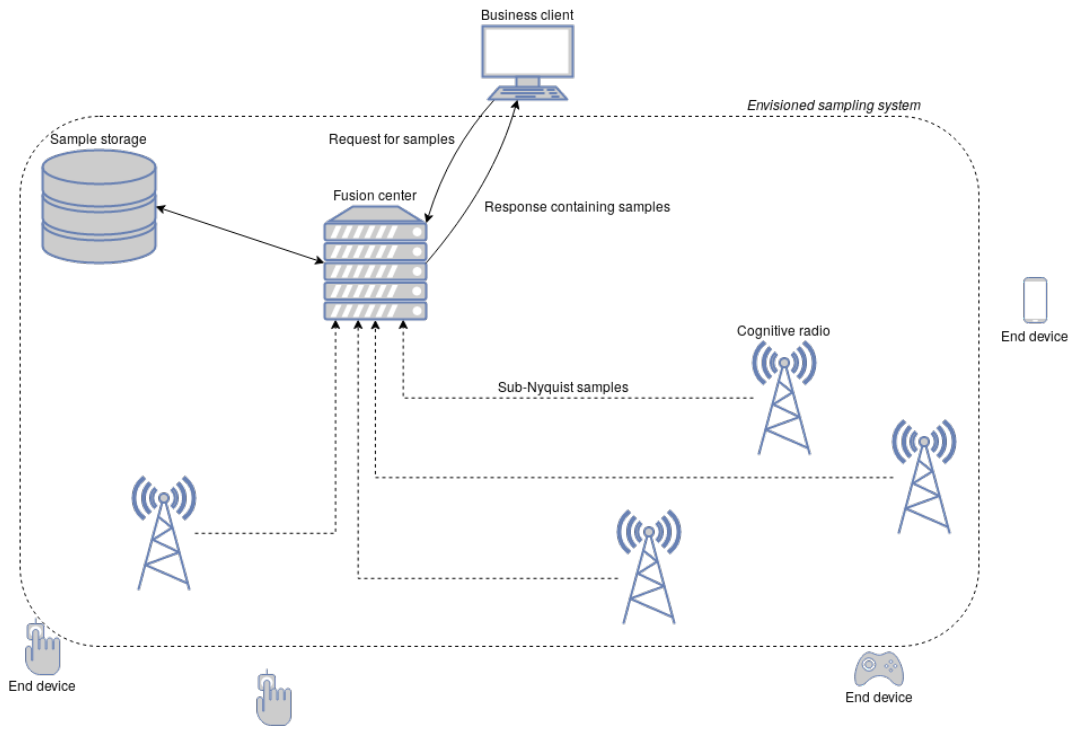


Figure 5.1: Proposed scenario of an IoT-aimed, multi-site, sub-Nyquist sampling infrastructure.

The first opportunity a multi-site infrastructure offers is the spatial diversity of the IoT messages. Indeed, having separate sampling locations can mitigate channel impairments caused by the geographical environment, *e.g.* shadowing. If a message from a device is not received by the only reception antenna of a mono-site sampling infrastructure because of shadowing, it is irremediably lost. On the contrary, in case the sampling infrastructure is multi-site, the message may be received by another antenna. This adds resilience and robustness to the sampling infrastructure. However, this added spatial diversity can be problematic for the step of support combination of the infrastructure. Indeed, in the mono-site MRS, presence of signal at some location (in the frequency spectrum) in one branch and absence of signal at the same location in

another branch results to the system considering the artefact as an alias instead of an original signal component. In the multi-site infrastructure, absence of signal in a branch can be caused by a non-reception of the signal by the reception site relative to this branch, leading actual signal components to be considered as aliases. To mitigate this issue, voting rules can be implemented. Early into this PhD, a peer-reviewed publication in an international conference focused on the benefits of spatial diversity for an MRS-based sampling infrastructure and investigated several voting rules [28]. However, the choice of the optimal voting rule - and especially how to set it *a priori* - remains an open question.

Another opportunity that a multi-site infrastructure can capitalize on is the signal sparsity in the spatial domain. It might be so that the density of IoT devices is uneven, with some reception sites of the sampling infrastructure surrounded by many IoT devices and some others surrounded by very few of them. In this case, the sites which receive little activity from IoT devices have the possibility to adapt their sampling settings accordingly. The decision to adapt to the radio environment can be taken at the reception site or at the fusion center. This additional domain of sparsity is a new lever towards the reduction of the number of acquired samples, and thus, towards an increased efficiency of sampling. Note that gains in sample volumes can be counterbalanced by the overhead that a multi-site infrastructure induces, mostly through sending the received samples to the fusion center *via* a backhaul link.

Clock synchronization is a challenge that is present in a mono-site infrastructure and exacerbated in a multi-site infrastructure. The important thing to know is the delay between the clocks of the different branches. In the mono-site infrastructure, a possible solution for clock synchronicity would be to have each branch's ADC controlled by a separate clock, with a separate low-rate master clock used to measure the drift between branches' clocks after each block (a duration introduced in Chapter 4). In the multi-site infrastructure, synchronization using GPS receivers is likely not to be sufficiently precise. However, a tight synchronization is only necessary for the recovery of the phase of the original signal, and not for previous steps such as support estimation and combination. A way to circumvent the issue of tight synchronization could then consist in using samples from the branches on different sites to estimate the signal support (as this step only requires a loose synchronization), then to use this estimate to reconstruct the signal phase from samples of different branches of the same reception site, for which tight synchronization can be assumed.

Appendix A

Appendix relative to Chapter 2

A.1 Relationship between P_D, η and $E_{\min}$ in Section 2.3.3.4

By definition of the cumulative distribution function (cdf), we have:

$$F(x; 2, 2E_{\min}) = P(2|Y_i|^2 < x). \tag{A.1}$$

From (2.16) and (A.1), we obtain:

$$F(2\eta; 2, 2E_{\min}) = P(2|Y_i|^2 < 2\eta) = 1 - P_D. \tag{A.2}$$

The cdf $F(x; k, \lambda)$ of the $\chi^2(k, \lambda)$ distribution is

$$F(x; k, \lambda) = e^{-\lambda/2} \sum_{j=0}^{+\infty} \frac{(\lambda/2)^j}{j!} Q\left(\frac{k}{2} + j, \frac{x}{2}\right), \tag{A.3}$$

where $Q(a, x)$ is the regularized lower incomplete gamma function, defined as:

$$Q(a, x) = \frac{1}{\Gamma(a)} \int_0^x t^{a-1} e^{-t} dt, \tag{A.4}$$

with $\Gamma(a)$ the gamma function.

Combining (A.2) and (A.3) yields:

$$1 - P_D = F(2\eta; 2, 2E_{\min}) = e^{-E_{\min}} \sum_{j=0}^{+\infty} \frac{{E_{\min}}^j}{j!} Q(1 + j, \eta), \tag{A.5}$$

leading to the relationship presented in (2.17).

Bibliography

[1] Youness Arjoune and Naima Kaabouch. A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions. *Sensors*, 19(1):126, jan 2019.

[2] David Makovoz. Noise variance estimation in signal processing. In *6th IEEE Int. Symp. Signal Process. and Inf. Technol.*, pages 364–369, 2006.

[3] Esteban Selva, Apostolos Kountouris, and Yves Louet. K-Means-Based Blind Noise Variance Estimation. In *IEEE Vehicular Technol. Conf. -Spring*, pages 1–7, 2021.

[4] Esteban Selva and Apostolos Kountouris. Détermination du bruit dans un signal, 2020.

[5] Esteban Selva, Apostolos Kountouris, and Yves Louet. On the Choice of Sampling Rates in Multi-Rate Sampling. In *Proc. Data Compression Conf.*, Snowbird, UT, USA, 2021.

[6] International Conference on Safety of Life at Sea. *Text of the Convention for the Safety of Life at Sea*. 1914.

[7] 69th Congress, United States Public Law 632. *Radio Act of 1927*. 1927.

[8] Theodore S. Rappaport. The wireless revolution. *IEEE Communications Magazine*, 29(11):52–71, 1991.

[9] Ghanshyam Singh and Viranjay M. Srivastava. *MOSFET Technologies for Double-Pole Four-Throw Radio-Frequency Switch*. Springer International Publishing, 2014.

[10] D. H. Ring. Cover Sheet for Technical Memoranda: Mobile Telephony - Wide Area Coverage. 1947.

[11] Mugen Peng, Yaohua Sun, Xuelong Li, Zhendong Mao, and Chonggang Wang. Recent Advances in Cloud Radio Access Networks: System Architectures, Key Techniques, and Open Issues. *IEEE Communications Surveys and Tutorials*, 18(3):2282–2308, 2016.

[12] Giovanni Interdonato. *Signal Processing Aspects of Cell-Free Massive MIMO*. PhD thesis, Linköping University, 2018.

[13] Joseph Mitola and Gerald Q. Maguire. Cognitive radio: Making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, 1999.

[14] Sreeraj Rajendran, Roberto Calvo-Palomino, Markus Fuchs, Bertold Van Den Bergh, Hector Cordobes, Domenico Giustiniano, Sofie Pollin, and Vincent Lenders. Electrosense: Open and Big Spectrum Data. *IEEE Communications Magazine*, 56(1):210–217, 2018.

[15] Tony Cai and Lie Wang. Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE Transactions on Information Theory*, 57(7):4680–4688, 2011.

[16] Hans Dieter Lüke. The Origins of the Sampling Theorem. *IEEE Communications Magazine*, 37(4):106–108, 1999.

[17] Brad Brannon and Jon Hall. Understanding State of the Art in ADCs. Technical report, Analog Devices, Inc., Norwood, MA, USA, 2008.

[18] Emmanuel Candès, Justin Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[19] David L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[20] Michael Fleyer, Alexander Linden, Moshe Horowitz, and Amir Rosenthal. Multirate synchronous sampling of sparse multiband signals. *IEEE Trans. Signal Process.*, 58(3):1144–1156, mar 2010.

[21] H. J. Landau. Necessary density conditions for sampling and interpolation of certain entire functions. *Acta Mathematica*, 117(1):37–52, 1967.

[22] Qing Zhao and Brian M. Sadler. A survey of dynamic spectrum access. *IEEE Signal Processing Magazine*, 24(3):79–89, 2007.

[23] Tevfik Yucek and Husey Arslam. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Commun. Surv.*, 11(1):116–130, mar 2009.

[24] Zhi Tian and Georgias B. Giannakis. Compressed sensing for wideband cognitive radios. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 4, 2007.

[25] Zhi Tian. Compressed Wideband Sensing in Cooperative Cognitive Radio Networks. *GLOBECOM - IEEE Global Telecommunications Conference*, pages 3756–3760, 2008.

[26] Ian F. Akyildiz, Brandon F. Lo, and Ravikumar Balakrishnan. Cooperative spectrum sensing in cognitive radio networks: A survey. *Physical Communication*, 4(1):40–62, 2011.

[27] Dyonisius Dony Ariananda and Geert Leus. Cooperative compressive wideband power spectrum sensing. *Conference Record - Asilomar Conference on Signals, Systems and Computers*, pages 303–307, 2012.

[28] Esteban Selva, Yves Louët, and Apostolos Kountouris. Sparse signal detection with spatial diversity using multi-rate sampling. In *URSI AP-RASC*, 2019.

[29] Niall Hurley and Scott Rickard. Comparing measures of sparsity. *IEEE Transactions on Information Theory*, 55(10):4723–4741, 2009.

[30] David S. Taubman and Michael W. Marcellin. *JPEG2000: Image Compression Fundamentals, Standards and Practice.* 2002.

[31] Nasir Ahmed, T Natarajan, and K. R. Rao. Discrete Cosine Transform. *IEEE Trans. Computers*, C-23(1):90–93, 1974.

[32] David L. Donoho. De-Noising by Soft-Thresholding. *IEEE Trans. Inf. Theor.*, 41(3):613–627, 1995.

[33] Ronald A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

[34] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1999.

[35] Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images, 1996.

[36] Y.M. Lu and M.N. Do. Sampling Signals from a Union of Subspaces. *IEEE Signal Process. Mag.*, 25(2):41–47, 2008.

[37] Martin Vetterli, Pina Marziliano, and Thierry Blu. Sampling Signals With Finite Rate of Innovation. *IEEE Trans. Signal Process.*, 50(6):1417–1428, 2002.

[38] Yonina C. Eldar and Gitta Kutyniok. *Compressed Sensing: Theory and Applications.* Cambridge University Press, 2012.

[39] David L. Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization. *Proc. Nat. Acad. Sci. USA*, 100(5):2197–2202, 2003.

[40] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24(2):227–234, 1995.

[41] Holger Rauhut. Compressive Sensing and Structured Random Matrices. 2010.

[42] Emmanuel Candès and Terence Tao. Decoding by Linear Programming. *IEEE Trans. Inf. Theory*, 51(12):4203–4215, 2005.

[43] Mark A. Davenport. *Random Observations on Random Observations : Sparse Signal Acquisition and Processing.* PhD thesis, 2010.

[44] Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via Orthogonal Matching Pursuit. *IEEE Trans. Inf. Theor.*, 53(12):4655–4666, 2007.

[45] Thomas Strohmer and Robert W. Heath. Grassmannian frames with applications to coding and communication. *Appl. and Comput. Harm. Anal.*, 14(3):257–275, 2003.

[46] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Compressed sensing and best k-term approximation. *J. Am. Math. Soc.*, 22(1):211–231, 2009.

[47] Matthew A. Herman and Thomas Strohmer. High-resolution radar via compressed sensing. *IEEE Trans. Signal Process.*, 57(6):2275–2284, 2009.

[48] Jean Bourgain, Stephen Dilworth, Kevin Ford, Sergei Konyagin, and Denka Kutzarova. Explicit constructions of rip matrices and related problems. *Duke Math. J.*, 159(1):145–185, 2011.

[49] Ronald A. DeVore. Deterministic constructions of compressed sensing matrices. *J. Complexity*, 23(4-6):918–925, 2007.

[50] Afonso S. Bandeira, Matthew Fickus, Dustin G. Mixon, and Percy Wong. The Road to Deterministic Matrices with the Restricted Isometry Property. *J. Fourier Anal. and Applications*, 19(6):1123–1149, 2013.

[51] Jelani Nelson, Eric Price, and Mary Wootters. New constructions of RIP matrices with fast multiplication and fewer rows. *Proc. Annu. ACM-SIAM Symp. Discrete Algorithms*, pages 1515–1528, 2014.

[52] David Gamarnik. Explicit Construction of RIP Matrices Is Ramsey-Hard. *Commun. Pure and Appl. Math.*, 73(9):2043–2048, 2020.

[53] Jason N. Laska, Petros T. Boufounos, Mark A. Davenport, and Richard G. Baraniuk. Democracy in action: Quantization, saturation, and compressive sensing. *Appl. and Comput. Harmon. Anal.*, 31(3):429–443, 2011.

[54] Richard G Baraniuk, M Davenport, R DeVore, and M B Wakin. A Simple Proof of the Restricted Isometry Property for Random Matrices (aka the Johnson-Lindenstrauss lemma meets compressed sensing). *Construct. Approx.*, 28:253–263, 2008.

[55] Alfred M. Bruckstein, David L. Donoho, and Michael Elad. From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images. *SIAM Review*, 51(1):34–81, 2009.

[56] David L. Donoho and Xiaoming Huo. Uncertainty Principles and Ideal Atomic Decomposition. *IEEE Trans. Inform. Theor.*, 47(7):2845–2862, 2001.

[57] David L. Donoho and Philip B. Stark. Uncertainty Principles and Signal Recovery. *SIAM J. App. Math.*, 49(3):906–931, 1989.

[58] David L. Donoho and Jared Tanner. Neighborliness of randomly projected simplices in high dimensions. *Proc. Nat. Acad. Sci. USA*, 102(27):9452–9457, 2005.

[59] David L. Donoho, Yaakov Tsaig, Iddo Drori, and Jean-Luc Starck. Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit. *IEEE Trans. Inform. Theor.*, 58(2):1094–1121, 2012.

[60] David L. Donoho, Iain Johnstone, and Andrea Montanari. Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising. *IEEE Trans. Inform. Theor.*, 59(6):3396–3433, 2013.

[61] Yihong Wu and Sergio Verdú. Optimal Phase Transitions in Compressed Sensing. *IEEE Trans. Inf. Theor.*, 58(10):6241–6263, 2012.

[62] S. Muthukrishnan. Data streams: Algorithms and applications. *Foundations and Trends in Theoretical Computer Science*, 1(2):117–236, 2005.

[63] Nikolas P. Galatsanos and Aggelos K. Katsaggelos. Methods for Choosing the Regularization Parameter and Estimating the Noise Variance in Image Restoration and Their Relation. *IEEE Trans. Image Process.*, 1(3):322–336, 1992.

[64] Gene H. Golub, Michael Heath, and Grace Wahba. Generalized Cross-Validation as a Method for Choosing a Good Ridge Parameter. *Technometrics*, 21(2):215–223, 1979.

[65] George B. Dantzig. Origins of the simplex method. Technical report, 1987.

[66] Amir Beck and Marc Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm. *SIAM J. Imag. Sci.*, 2(1):183–202, 2009.

[67] Mário A.T. Figueiredo, Robert D. Nowak, and Stephen J. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J. Selected Topics Signal Process.*, 1(4):586–597, 2007.

[68] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1–22, 2010.

[69] Elaine T. Hale, Wotao Yin, and Yin Zhang. Fixed-point continuation for 11-minimization: Methodology and convergence. *SIAM J. Optim.*, 19(3):1107–1130, 2008.

[70] F. Malgouyres and T. Zeng. A predual proximal point algorithm solving a non negative basis pursuit denoising model. *Int. J. Comput. Vision*, 83(3):294–311, 2009.

[71] Seung-Jean Kim, Kwangmoo Koh, Michael Lustig, and Stephen Boyd. An efficient method for Compressed Sensing. *IEEE Int. Conf. Imag. Process.*, pages 117–120, 2007.

[72] Bob L. Sturm and Mads Græsbøll Christensen. Comparison of Orthogonal Matching Pursuit Implementations. *Proc. Eur. Signal Process. Conf.*, pages 220–224, 2012.

[73] Anmin Huang, Gui Guan, Qun Wan, and Abolfazl Mehbodniya. A block orthogonal matching pursuit algorithm based on sensing dictionary. *Int. J. Phys. Sci.*, 6(5):992–999, 2011.

[74] Deanna Needell and Roman Vershynin. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.*, 9(3):317–334, 2009.

[75] Deanna Needell and Joel A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Commun. ACM*, 53(12):93–100, 2010.

[76] Thomas Blumensath and Mike E. Davies. Iterative hard thresholding for compressed sensing. *Appl. and Comput. Harm. Anal.*, 27(3):265–274, 2009.

[77] Thomas Blumensath and Mike E. Davies. Iterative thresholding for saparse approximations. *J. Fourier Anal. and Applications*, 14(5-6):629–654, 2008.

[78] T Blumensath and M E Davies. How To Use the Iterative Hard Thresholding Algorithm. *Signal Processing*, 2009.

[79] Susanne Sparrer and Robert F.H. Fischer. Enhanced iterative hard thresholding for the estimation of discrete-valued sparse signals. *Eur. Signal Process. Conf.*, pages 71–75, 2016.

[80] Xiaobo Zhang, Wenbo Xu, Jiaru Lin, and Yifei Dang. Block normalised iterative hard thresholding algorithm for compressed sensing. *Electronics Letters*, 55(17):957–959, 2019.

[81] Ding-Zhu Du and Frank K. Hwang. *Combinatorial Group Testing and Its Applications*, volume 12. 2000.

[82] Sabyasachi Ghosh, Rishi Agarwal, Mohammad Ali Rehan, Shreya Pathak, Pratyush Agrawal, Yash Gupta, Sarthak Consul, Nimay Gupta, Ritika Goyal, Ajit Rajwade, and Manoj Gopalkrishnan. A compressed sensing approach to group-testing for covid-19 detection. *arXiv*, pages 1–15, 2020.

[83] George K. Atia and Venkatesh Saligrama. Boolean compressed sensing and noisy group testing. *IEEE Trans. Inf. Theor.*, 58(3):1880–1901, 2012.

[84] Anna Gilbert and Piotr Indyk. Sparse recovery using sparse matrices. *Proc. IEEE*, 98(6):937–947, 2010.

[85] Shihao Ji, Ya Xue, and Lawrence Carin. Bayesian compressive sensing. *IEEE Trans. Signal Process.*, 56(6):2346–2356, 2008.

[86] Ljubiša Stanković, Ervin Sejdić, Srdjan Stanković, Miloš Daković, and Irena Orović. A Tutorial on Sparse Signal Reconstruction and Its Applications in Signal Processing. *Circuits, Systems, and Signal Processing*, 38(3):1206–1263, 2019.

[87] S. Derin Babacan, Rafael Molina, and Aggelos K. Katsaggelos. Bayesian Compressive Sensing Using Laplace Priors. *IEEE Trans. Image Process.*, 19(1):53–63, 2010.

[88] Tao Jiang, Xiao Wei Zhang, and Yingsong Li. Bayesian compressive sensing using reweighted laplace priors. *AEU - Int. J. Electron. and Commun.*, 97:178–184, 2018.

[89] Vangelis P. Oikonomou, Spiros Nikolopoulos, and Ioannis Kompatsiaris. A novel compressive sensing scheme under the variational Bayesian framework. *Eur. Signal Process. Conf.*, pages 9–13, 2019.

[90] Stéphanie Bidon, Olivier Besson, Jean Yves Tourneret, and François Le Chevalier. Bayesian sparse estimation of migrating targets in autoregressive noise for wideband radar. *IEEE Nat. Radar Conf.*, 50(2):579–584, 2014.

[91] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM*, pages 84–90, 2017.

[92] Zubair Md Fadlullah, Fengxiao Tang, Bomin Mao, Nei Kato, Osamu Akashi, Takeru Inoue, and Kimihiro Mizutani. State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems. *IEEE Commun. Surv. and Tut.*, 19(4):2432–2455, 2017.

[93] Akanksha Rai Sharma and Pranav Kaushik. Literature survey of statistical, deep and reinforcement learning in natural language processing. *IEEE Int. Conf. Comput. Commun. Automat.*, pages 350–354, 2017.

[94] Huazhe Xu, Yang Gao, Fisher Yu, and Trevor Darrell. End-to-end learning of driving models from large-scale video datasets. *IEEE Conf. Computer Vision and Pattern Recognition*, 2017:3530–3538, 2017.

[95] Ali Mousavi, Ankit B. Patel, and Richard G. Baraniuk. A deep learning approach to structured signal recovery. *Annu. Allerton Conf. Commun. Control Comput.*, pages 1336–1343, 2016.

[96] Daisuke Ito, Satoshi Takabe, and Tadashi Wadayama. Trainable ISTA for Sparse Signal Recovery. *IEEE Trans. Signal Process.*, 67(12):3113–3125, 2019.

[97] E. C. Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang. Deep learning approaches for sparse recovery in compressive sensing. *Int. Symp. Image Signal Process. Anal., ISPA*, (2):129–134, 2019.

[98] Youness Arjoune, Naima Kaabouch, Hassan El Ghazi, and Ahmed Tamtaoui. Compressive sensing: Performance comparison of sparse recovery algorithms. *IEEE Annu. Comput. and Commun. Workshop and Conf.*, pages 1–7, 2017.

[99] Hengyong Yu and Ge Wang. Compressed sensing based interior tomography. *Phys. Med. Biol.*, 54(13):4341, 2009.

[100] Y. Wiaux, L. Jacques, G. Puy, A. M.M. Scaife, and P. Vandergheynst. Compressed sensing imaging techniques for radio interferometry. *Mon. Not. R. Astron. Soc*, 395(3):1733–1742, 2009.

[101] Marco F. Duarte, Mark A. Davenport, Dharmpal Takhar, Jason N. Laska, Ting Sun, Kevin F. Kelly, and Richard G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Process. Mag.*, 25(2):83–91, 2008.

[102] Jun Won Choi, Byonghyo Shim, Yacong Ding, Bhaskar Rao, and Dong In Kim. Compressed Sensing for Wireless Communications: Useful Tips and Tricks. *IEEE Communications Surveys and Tutorials*, 19(3):1527–1550, 2017.

[103] Afif Osseiran, Federico Boccardi, Volker Braun, Katsutoshi Kusume, Patrick Marsch, Michal Maternia, Olav Queseth, Malte Schellmann, Hans Schotten, Hidekazu Taoka, Hugo Tullberg, Mikko A. Uusitalo, Bogdan Timus, and Mikael Fallgren. Scenarios for 5G mobile and wireless communications: The vision of the METIS project. *IEEE Communications Magazine*, 52(5):26–35, 2014.

[104] Tarik Taleb and Andreas Kunz. Machine type communications in 3GPP networks: Potential, challenges, and solutions. *IEEE Communications Magazine*, 50(3):178–184, 2012.

[105] Reza Hoshyar, Ferry P. Wathan, and Rahim Tafazolli. Novel low-density signature for synchronous CDMA systems over AWGN channel. *IEEE Transactions on Signal Processing*, 56(4):1616–1626, 2008.

[106] Christian R. Berger, Zhaohui Wang, Jianzhong Huang, and Shengli Zhou. Application of Compressive Sensing to Sparse Channel Estimation. *IEEE Communications Magazine*, 48(11):164–174, 2010.

[107] Akbar M. Sayeed. Deconstructing multiantenna fading channels. *IEEE Transactions on Signal Processing*, 50(10):2563–2579, 2002.

[108] Jose L. Paredes, Gonzalo R. Arce, and Zhongmin Wang. Ultra-wideband compressed sensing: Channel estimation. *IEEE Journal on Selected Topics in Signal Processing*, 1(3):383–395, 2007.

[109] Mustafa Riza Akdeniz, Yuanpeng Liu, Mathew K. Samimi, Shu Sun, Sundeep Rangan, Theodore S. Rappaport, and Elza Erkip. Millimeter wave channel modeling and cellular capacity evaluation. *IEEE Journal on Selected Areas in Communications*, 32(6):1164–1179, 2014.

[110] Jarvis Haupt, Waheed U. Bajwa, Gil Raz, and Robert Nowak. Toeplitz compressed sensing matrices with applications to sparse channel estimation. *IEEE Transactions on Information Theory*, 56(11):5862–5875, 2010.

[111] Kazunori Hayashi, Masaaki Nagahara, and Toshiyuki Tanaka. A User's Guide to Compressed Sensing for Communications. *IEICE Trans. Commun.*, E96-B(3):685–712, 2013.

[112] Saad Qaisar, Rana Muhammad Bilal, Wafa Iqbal, Muqaddas Naureen, and Sungyoung Lee. Compressive sensing: From theory to applications, a survey. *J. Commun. and Netw.*, 15(5):443–456, 2013.

[113] Amir Rosenthal, Alexander Linden, and Moshe Horowitz. Multirate asynchronous sampling of sparse multiband signals. *J. Opt. Soc. Am. A*, 25(9):2320–2330, 2008.

[114] Marguerite Marnat. *Radiofrequency receivers based on compressive sampling for feature extraction in cognitive radio applications.* PhD thesis, 2018.

[115] Deborah Cohen, Shahar Tsiper, and Yonina C. Eldar. Analog-to-Digital Cognitive Radio: Sampling, Detection, and Hardware. *IEEE Signal Process. Mag.*, 35(1):137–166, jan 2018.

[116] Raman Venkataramani and Yoram Bresler. Perfect reconstruction formulas and bounds on aliasing error in sub-Nyquist nonuniform sampling of multiband signals. *IEEE Transactions on Information Theory*, 46(6):2173–2183, 2000.

[117] Moshe Mishali and Yonina C. Eldar. Blind multiband signal reconstruction: Compressed sensing for analog signals. *IEEE Transactions on Signal Processing*, 57(3):993–1009, 2009.

[118] M. E. Domínguez-Jiménez, N. González-Prelcic, G. Vazquez-Vilar, and R. López-Valcarce. Design of universal multicoset sampling patterns for compressed sensing of multiband sparse signals. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 3337–3340, 2012.

[119] Samba Traore. *Contribution à l'étude de l'échantillonnage non uniforme dans le domaine de la radio intelligente.* PhD thesis, CentraleSupélec Ecole Doctorale MATISSE, 2016.

[120] Ping Feng and Yoram Bresler. Spectrum-blind minimum-rate sampling and reconstruction of multiband signals. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, 1996.

[121] Michael Wakin, Stephen Becker, Eric Nakamura, Michael Grant, Emilio Sovero, Daniel Ching, Juhwan Yoo, Justin Romberg, Azita Emami-Neyestanak, and Emmanuel Candes. A Nonuniform sampler for wideband spectrally-sparse environments. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2(3):516–529, 2012.

[122] Moshe Mishali and Yonina C Eldar. Sub-nyquist sampling: Bridging theory and practice. *IEEE Signal Processing Magazine*, 28(6):98–124, 2011.

[123] Linda Bai and Sumit Roy. Compressive spectrum sensing using a bandpass sampling architecture. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2(3):433–442, 2012.

[124] Hongjian Sun, Wei Yu Chiu, Jing Jiang, Arumugam Nallanathan, and H. Vincent Poor. Wideband spectrum sensing with sub-nyquist sampling in cognitive radios. *IEEE Transactions on Signal Processing*, 60(11):6068–6073, 2012.

[125] Moshe Mishali and Yonina C. Eldar. From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals. *IEEE Journal on Selected Topics in Signal Processing*, 4(2):375–391, 2010.

[126] Jason N. Laska, Sami Kirolos, Marco F. Duarte, Tamer S. Ragheb, Richard G. Baraniuk, and Yehia Massoud. Theory and Implementation of an Analog-to-Information Converter using Random Demodulation. *2007 IEEE International Symposium on Circuits and Systems*, pages 1959–1962, 2007.

[127] Yilun Chen, Moshe Mishali, Yonina C. Eldar, and Alfred O. Hero. Modulated wideband converter with non-ideal lowpass filters. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 3630–3633, 2010.

[128] B. Razavi. A 60-GHz CMOS receiver front-end. *IEEE Journal of Solid-State Circuits*, 41(1):17–22, 2006.

[129] Stephen Becker. Practical Compressed Sensing: modern data acquisition and signal processing. *Thesis*, 2011, 2011.

[130] Marguerite Marnat, Michael Pelissier, Olivier Michely, and Laurent Ros. Code properties analysis for the implementation of a modulated wideband converter. *25th European Signal Processing Conference, EUSIPCO 2017*, 2017-Janua:2121–2125, 2017.

[131] Jefferson Ribadeneira-Ramírez, Gerardo Martínez, David Gómez-Barquero, and Narcís Cardona. Interference analysis between digital terrestrial television (DTT) and 4G LTE mobile networks in the digital dividend bands. *IEEE Transactions on Broadcasting*, 62(1):24–34, 2016.

[132] Michael Marcus and Bruno Pattan. Millimeter Wave Propagation: Spectrum Management Implications. *IEEE Microwave Magazine*, 6(2):54–62, 2005.

[133] Václav Valenta, Roman Maršálek, Geneviève Baudoin, Martine Villegas, Martha Suarez, and Fabien Robert. Survey on spectrum utilization in Europe: Measurements, analyses and observations. *2010 Proceedings of the 5th International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CROWNCom 2010*, 2010.

[134] M. Mehdawi, N. G. Riley, M. Ammar, A. Fanan, and M. Zolfaghari. Spectrum occupancy measurements and lessons learned in the context of cognitive radio. *2015 23rd Telecommunications Forum, TELFOR 2015*, pages 196–199, 2016.

[135] Karsten Buckwitz, Jan Engelberg, and Gernot Rausch. Licensed Shared Access (LSA) - Regulatory background and view of Administrations. *Proceedings of the 2014 9th International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CROWNCOM 2014*, pages 413–416, 2014.

[136] "Citizens Broadband Radio Service", 47 C.F.R § 96. 2016.

[137] Hongjian Sun, Arumugam Nallanathan, Cheng Xiang Wang, and Yunfei Chen. Wideband spectrum sensing for cognitive radio networks: A survey. *IEEE Wireless Communications*, 20(2):74–81, 2013.

[138] Steven M Kay. *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory.* Prentice Hall, Upper Saddle River, NJ, USA, 1998.

[139] Harry Urkowitz. Energy Detection of Unknown Deterministic Signals. *Proceedings of the IEEE*, 55(4):523–531, 1967.

[140] Eva Lagunas, Shree Krishna Sharma, Symeon Chatzinotas, and Björn Ottersten. Compressive Sensing based Energy Detector. *European Signal Processing Conference*, 2016-Novem:1678–1682, 2016.

[141] Jonti Talukdar, Bhavana Mehta, Kinjal Aggrawal, and Mansi Kamani. Implementation of SNR estimation based energy detection on USRP and GNU radio for cognitive radio networks. In *Proc. 2017 Int. Conf. Wireless Commun., Signal Process. and Netw.*, pages 304–308, 2017.

[142] Deepak R. Joshi, Dimitrie C. Popescu, and Octavia A. Dobre. Adaptive spectrum sensing with noise variance estimation for dynamic cognitive radio systems. *2010 44th Annual Conference on Information Sciences and Systems, CISS 2010*, (1), 2010.

[143] Jinbo Wu, Tao Luo, and Guangxin Yue. An energy detection algorithm based on double-threshold in cognitive radio systems. *2009 1st International Conference on Information Science and Engineering, ICISE 2009*, pages 493–496, 2009.

[144] Jérôme Antoni. Cyclic spectral analysis in practice. *Mechanical Systems and Signal Processing*, 21(2):597–630, 2007.

[145] Amod V. Dandawaté and Georgios B. Giannakis. Statistical Tests for Presence of Cyclostationarity. *IEEE Transactions on Signal Processing*, 42(9):2355–2369, 1994.

[146] Ziad Khalaf. *Contributions à l'étude de détection des bandes libres dans le contexte de la radio intelligente.* PhD thesis, CentraleSupelec, 2013.

[147] Shree Krishna Sharma, Tadilo Endeshaw Bogale, Symeon Chatzinotas, Long Bao Le, Xianbin Wang, and Bjorn Ottersten. Improving robustness of cyclostationary detectors to cyclic frequency mismatch using Slepian basis. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, 2015-Decem:456–460, 2015.

[148] Xinzhi Zhang, Rong Chai, and Feifei Gao. Matched filter based spectrum sensing and power level detection for cognitive radio network. *2014 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2014*, pages 1267–1270, 2014.

[149] Fatima Salahdine, Hassan El Ghazi, Naima Kaabouch, and Wassim Fassi Fihri. Matched filter detection with dynamic threshold for cognitive radio networks. *International Conference on Wireless Networks and Mobile Communications, WINCOM 2015*, pages 2–7, 2016.

[150] Yonghong Zeng and Ying Chang Liang. Spectrum-sensing algorithms for cognitive radio based on statistical covariances. *IEEE Transactions on Vehicular Technology*, 58(4):1804–1815, 2009.

[151] Yonghong Zeng and Ying Chang Liang. Eigenvalue-based spectrum sensing algorithms for cognitive radio. *IEEE Trans. Commun.*, 57(6):1784–1793, 2009.

[152] Boaz Nadler, Federico Penna, and Roberto Garello. Performance of eigenvalue-based signal detectors with known and unknown noise level. *IEEE Int. Conf. Commun.*, pages 4–8, 2011.

[153] P. Bianchi, M. Debbah, M. Maida, and J. Najim. Performance of statistical tests for single-source detection using random matrix theory. *IEEE Trans. Inf. Theor.*, 57(4):2400–2419, 2011.

[154] Mohamed Hamid, Niclas Björsell, and Slimane Ben Slimane. Sample covariance matrix eigenvalues based blind SNR estimation. *IEEE Instrum. and Meas. Technol. Conf.*, pages 718–722, 2014.

[155] Said E. El-Khamy, Mohamed S. El-Mahallawy, and Youssef S. El-Nasser. Improved Wide-band Spectrum Sensing Techniques Using Wavelet-Based Edge Detection for Cognitive Radio. In *2013 Int. Conf. Comput. Netw. and Commun.*, pages 418–423, 2013.

[156] Zhi Quan, Shuguang Cui, Ali H. Sayed, and H. Vincent Poor. Optimal multiband joint detection for spectrum sensing in cognitive radio networks. *IEEE Trans. Signal Process.*, 57(3):1128–1140, 2009.

[157] I. Raghu, Sai Sumanth Chowdary, and Elizabeth Elias. Efficient spectrum sensing for Cognitive Radio using Cosine Modulated Filter Banks. *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, pages 2086–2089, 2017.

[158] Jun Li. *A Wavelet Approach to Edge Detection.* Master of science, Sam Houston State University, 2003.

[159] Shree Krishna Sharma, Eva Lagunas, Symeon Chatzinotas, and Bjorn Ottersten. Application of compressive sensing in cognitive radio communications: A survey. *IEEE Communications Surveys and Tutorials*, 18(3):1838–1860, 2016.

[160] Mark A. Davenport, Marco F. Duarte, Michael B. Wakin, Jason N. Laska, Dharmpal Takhar, Kevin F. Kelly, and Richard G. Baraniuk. The smashed filter for compressive classification and target recognition. *Proc. SPIE Computational Imaging V*, 2007.

[161] Eva Lagunas and Luca Rugini. Performance of compressive sensing based energy detection. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, 2017-Octob:1–5, 2018.

[162] Linxiao Yang, Jun Fang, Huiping Duan, and Hongbin Li. Fast compressed power spectrum estimation: Towards a practical solution for wideband spectrum sensing. *IEEE Trans. Wireless Commun.*, 19(1):520–532, 2020.

[163] Zhi Tian, Yohannes Tafesse, and Brian M. Sadler. Cyclic feature detection with sub-nyquist sampling for wideband spectrum sensing. *IEEE Journal on Selected Topics in Signal Processing*, 6(1):58–69, 2012.

[164] Deborah Cohen and Yonina C Eldar. Compressed cyclostationary detection for Cognitive Radio. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 3509–3513, 2017.

[165] Youness Arjoune and Naima Kaabouch. Wideband spectrum sensing: A bayesian compressive sensing approach. *Sensors (Switzerland)*, 18(6), 2018.

[166] Thakshila Wimalajeewa and Pramod K. Varshney. Sparse Signal Detection With Compressive Measurements via Partial Support Set Estimation. *IEEE Transactions on Signal and Information Processing over Networks*, 3(1):46–60, 2017.

[167] Ahmed Al Hilli, Laleh Najafizadeh, and Athina P. Petropulu. A Weighted Approach for Sparse Signal Support Estimation with Application to EEG Source Localization. *IEEE Trans. Signal Process.*, 65(24):6551–6565, 2017.

[168] Youye Xie, Michael B. Wakin, and Gongguo Tang. Support recovery for sparse signals with non-stationary modulation. *arXiv preprint arXiv:1910.13104*, pages 1–1, 2019.

[169] Yonghua Wang, Yongwei Zhang, Pin Wan, Shunchao Zhang, and Jian Yang. A Spectrum Sensing Method Based on Empirical Mode Decomposition and K-Means Clustering Algorithm. *Wireless Commun. and Mobile Comput.*, 2018, 2018.

[170] Libin K. Mathew, Shreejith Shanker, A. P. Vinod, and A. S. Madhukumar. An Adaptive Energy Detection Scheme with Real-Time Noise Variance Estimation. *Circuits, Syst., and Signal Process.*, 39(5):2623–2647, 2020.

[171] Youness Arjoune, Zakaria El Mrabet, Hassan El Ghazi, and Ahmed Tamtaoui. Spectrum sensing: Enhanced energy detection technique based on noise measurement. In *2018 IEEE 8th Annu. Comput. and Commun. Workshop and Conf.*, pages 828–834, 2018.

[172] Pankaj Verma and Brahmjit Singh. Threshold Optimization in Energy Detection Scheme for Maximizing the Spectrum Utilization. *Procedia Comput. Sci.*, 93:191–198, 2016.

[173] Vincent Savaux. Detector based on the energy of filtered noise. *IET Signal Process.*, 13(1):36–45, 2019.

[174] David R. Pauluzzi and Norman C. Beaulieu. A comparison of SNR estimation techniques for the AWGN channel. *IEEE Trans. Commun.*, 48(10):1681–1691, 2000.

[175] J. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. *5th Berkeley Symp.*, 25(8), 1967.

[176] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, Hoboken, NJ, USA, 2005.

[177] Stuart P. Lloyd. Least Squares Quantization in PCM. *IEEE Trans. Inf. Theor.*, 28(2):129–137, 1982.

[178] David Arthur and Sergei Vassilvitskii. K-means++: The advantages of careful seeding. *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007.

[179] J. A. Hartigan and M. A. Wong. Algorithm AS 136: A K-Means Clustering Algorithm. *J. Roy. Statis. Soc. Ser. C (App. Statis.)*, 28(1):100, 1979.

[180] Erik W Grafarend. *Linear and nonlinear models: fixed effects, random effects, and mixed models.* Walter de Gruyter, 2007.

[181] Galen Reeves and Michael Gastpar. Sampling bounds for sparse support recovery in the presence of noise. *IEEE Int. Symp. Inf. Theor.*, pages 2187–2191, 2008.

[182] Jonathan Scarlett and Volkan Cevher. Limits on Support Recovery with Probabilistic Models: An Information-Theoretic Framework. *IEEE Int. Symp. Inf. Theor.*, 2015.

[183] Kae Won Choi and Ekram Hossain. Opportunistic access to spectrum holes between packet bursts: A learning-based approach. *IEEE Transactions on Wireless Communications*, 10(8):2497–2509, 2011.

[184] Mohammed Mehdi Saleh and Hemalatha Rallapall. Quick Detection and Assignment of Spectrum Hole in Cognitive Radio. *Int. Conf. Intell. Syst., Data Mining and Inf. Technol.*, pages 124–127, 2014.

[185] Amir Ghasemi and Elvino S. Sousa. Collaborative spectrum sensing for opportunistic access in fading environments. *IEEE Int. Symp. New Frontiers in Dyn. Spectr. Access Netw.*, pages 131–136, 2005.

[186] Emmanuel J. Candes and Michael B. Wakin. An introduction to compressive sampling. *IEEE Signal Process. Mag.*, 25(2):21–30, mar 2008.

[187] Palghat P. Vaidyanathan and Piya Pal. Sparse Sensing with Co-Prime Samplers and Arrays. *IEEE Trans. Signal Process.*, 59(2):573–586, feb 2011.

[188] Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17:1–5, 2016.

[189] Ron Rubinstein, Michael Zibulevsky, and Michael Elad. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. *CS Technion*, pages 1–15, 2008.

DOCTORAT BRETAGNE LOIRE / MATHSTIC

CentraleSupélec

Titre : Numérisation, compression et reconstruction d'un trafic radio large échelle pour l'Internet des Objets

Mots clés : traitement de signal parcimonieux, Echantillonnage Comprimé, infrastructure d'échantillonnage, Internet des Objets.

Résumé : La multiplication des cas d'utilisation dans le cadre de l'Internet des Objets appelle au développement de nouveaux protocoles de télécommunications et de nouvelles modulations. Cependant, la nécessité de déployer une infrastructure dédiée à la collecte et au traitement des données envoyées par les appareils connectés peut freiner le développement de nouveaux standards de télécommunications. Dans cette thèse, nous proposons et étudions une infrastructure ne dépendant pas de la technologie de communication choisie, dont le but est de collecter et traiter les données émises par des appareils connectés déployés sur le terrain. Comme les communications des appareils connectés peuvent être sporadiques, notamment dans le cas des réseaux de capteurs, nous considérons que les signaux reçus par les points d'accès de l'infrastructure sont parcimonieux en temps et/ou en fréquence. Sous cette condition, le cadre de l'Echantillonnage Comprimé (EC) offre la possibilité d'abaisser le taux d'échantillonnage, par rapport au taux d'échantillonnage de Nyquist standard.
Pour cela, notre infrastructure proposée de collecte de données s'appuie sur l'Echantillonneur Multi-Taux (EMT), un schéma d'échantillonnage fondé sur l'EC qui permet la réduction du taux d'échantillonnage et la relaxation de contraintes matérielles liées à l'échantillonnage à haute fréquence. Dans cette thèse, nous proposons une analyse poussée afin de dimensionner de manière adéquate une infrastructure fondée sur l'EMT. Parmi les résultats majeurs se trouvent le paramétrage, à l'aide de nombres premiers entre eux, des taux d'échantillonnage de l'EMT, l'estimation de la variance du bruit et du support spectral fréquentiel pour des signaux parcimonieux en fréquence, et une analyse complète des performances de l'EMT en fonction de plusieurs paramètres (nombre d'échantillonneurs, niveau de parcimonie du signal, taux d'erreur binaire, et autres).

Title: Digitization, Compression and Reconstruction of a Large-Scale Radio Traffic for the Internet of Things

Keywords: sparse signal processing, Compressed Sensing, Multi-Rate Sampler, sampling infrastructure, Internet of Things.

Abstract: The multiplicity of novel use cases for the Internet of Things (IoT) calls for new telecommunications protocols and modulations. However, the necessity to roll out a dedicated infrastructure to collect and process data sent by IoT devices can hinder the development of new standards for communications. In this thesis, we propose and study a technology-agnostic infrastructure aimed at collecting and processing information transmitted by IoT devices deployed on the field. Because communications from IoT devices can be sporadic, in particular when sensor networks are concerned, we consider that the signals received by the infrastructure access points are sparse in the time and/or frequency domains. Under such a condition, the Compressed Sensing (CS) framework offers the possibility to lower the sampling rate from the standard Nyquist rate. Our proposed infrastructure for data collection relies on the Multi-Rate Sampler (MRS), a CS-based sampling scheme that permits the reduction of the sampling rate and a relaxation of hardware constraints related to high-speed sampling. In this thesis, we provide a comprehensive analysis to adequately size an MRS-based infrastructure. Major results include a provision based on coprime integers to define the sampling rates of the MRS, the estimation of the noise variance and of the frequency-domain spectrum support for frequency-sparse signals, and a complete analysis of the MRS performance depending on a variety of parameters (number of samplers, signal sparsity level, bit error rate, and others).