

APPROXIMATE MESSAGE PASSING ALGORITHMS FOR
COMPRESSED SENSING

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL
ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Arian Maleki
September 2011

© Copyright by Arian Maleki 2011
All Rights Reserved

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Prof. David L. Donoho) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Prof. Robert M. Gray)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Prof. Andrea Montanari)

Approved for the University Committee on Graduate Studies.

Preface

Compressed sensing refers to a growing body of techniques that ‘undersample’ high-dimensional signals and yet recover them accurately. Such techniques make fewer measurements than traditional sampling theory demands: rather than sampling proportional to frequency bandwidth, they make only as many measurements as the underlying ‘information content’ of those signals. However, as compared with traditional sampling theory, which can recover signals by applying simple linear reconstruction formulas, the task of signal recovery from reduced measurements requires nonlinear, and so far, relatively expensive reconstruction schemes. One popular class of reconstruction schemes uses linear programming (LP) methods; there is an elegant theory for such schemes promising large improvements over ordinary sampling rules in recovering sparse signals. However, solving the required LPs is substantially more expensive in applications than the linear reconstruction schemes that are now standard. In certain imaging problems, the signal to be acquired may be an image with 10^6 pixels and the required LP would involve tens of thousands of constraints and millions of variables. Despite advances in the speed of LP, such methods are still dramatically more expensive to solve than we would like. In this thesis we focus on a class of low computational complexity algorithms known as iterative thresholding. We study them both theoretically and empirically. We will also introduce a new class of algorithms called approximate message passing or AMP. These schemes have several advantages over the classical thresholding approaches. First, they take advantage of the statistical properties of the problem to improve the convergence rate and predictability of the algorithm. Second, the nice properties of these algorithms enable us to make very accurate theoretical predictions on the asymptotic performance of

LPs as well. It will be shown that more traditional techniques such as coherence and restricted isometry property are not able to make such precise predictions.

Notational Conventions

Notation	Description
\mathbb{R}	real numbers
\mathbb{R}^N	vector space of real valued N dimensional vectors
x_i	i^{th} element of vector x
$ \cdot $	if applied to a number, absolute value
$ \cdot $	if applied to a set, cardinality of the set
$\langle x, y \rangle$	$\sum_i x_i y_i$
$\langle x \rangle$	$\frac{1}{N} \sum_{i=1}^N x_i$
$\ x\ _p$	for a vector ℓ_p -norm or ℓ_p seminorm defined as $(\sum_i x_i ^p)^{\frac{1}{p}}$
$\ x\ _0$	ℓ_0 -norm of a vector. Number of nonzero elements in x
ℓ_p^∞	Banach space of all vectors in \mathbb{R}^∞ with bounded ℓ_p -norm
$B_p^N(R)$	ball of radius R in ℓ_p space
S^{N-1}	unit ℓ_2 sphere in \mathbb{R}^N
\mathbb{I}	indicator function
$(a)_+$	a if $a > 0$ and zero otherwise
$\text{sgn}(\cdot)$	sign of a number
$\eta(a; \lambda)$	soft thresholding function, i.e. $\text{sgn}(a)(a - \lambda)_+$
$\eta^H(a; \lambda)$	hard thresholding function, i.e. $a\mathbb{I}(a > \lambda)$
Υ_k^N	The set of all k -sparse, N -dimensional vectors
$N_\epsilon(T)$	minimum number of elements in an ϵ -covering of T

Notation	Description
n	number of measurements
N	dimension of original vector
k	sparsity level or the number of nonzeros of the original vector
$\ A\ _F$	Frobenius norm or Hilbert Schmidt norm of a matrix: $\sqrt{\sum_{i,j} A_{i,j}^2}$
$\ A\ _{p \rightarrow q}$	operator norm of a matrix from ℓ_p to ℓ_q
A^*	transpose or adjoint or hermitian of matrix A
A_i	i^{th} column of matrix A
A_{ij}	ij^{th} element of matrix A
A_J	submatrix of A with columns restricted to set J
$\sigma_{\max}(A)$	maximum singular value of matrix A
$\sigma_{\min}(A)$	minimum singular value of matrix A
I_n	identity matrix of size n
$\mathbb{P}(\cdot)$	probability of an event
$\mathbb{E}(\cdot)$	expected value of a random variable
$\ X\ _p$	for a random variable, $[\mathbb{E}(X^p)]^{\frac{1}{p}}$
$\mathcal{N}(\mu, \Sigma)$	Gaussian with mean μ and covariance Σ
$\phi(z)$	$\frac{e^{-z^2/2}}{\sqrt{2\pi}}$
$\Phi(z)$	$\int_{-\infty}^z \frac{e^{-t^2/2}}{\sqrt{2\pi}} dt$
$\text{logit}(\pi)$	$\log\left(\frac{\pi}{1-\pi}\right)$
$\mathcal{F}_{\epsilon, \gamma}$	family of distributions with $F(0^+) - F(0^-) > 1 - \epsilon$, $\mathbb{E}_F(\mu^2) \leq \epsilon\gamma^2$

Acknowledgements

During my PhD I benefited significantly from interaction with the people at Stanford. I would like to take this opportunity to thank them.

First, I wish to express my appreciation and gratitude to my advisor, Professor David L. Donoho. Dave's thoughtful discussions and careful criticisms have greatly influenced my thought in the last several years. His vision and creativeness has inspired this research. His high scientific standards and his enthusiasm for solving seemingly unsolvable problems were the main deriving forces that made this thesis possible. I am also indebted to Dave for his endless patience, generous support, and constant guidance.

I have been fortunate to have enjoyed the support and collaboration of Professor Gunnar Carlsson. Gunnar was the first person who introduced me to the scientific world and gave me strong motivation, support, and guidance. He spent much time to teach me valuable lessons in both research and life. I owe great thanks to him.

This work could not be possible without the collaboration with Professor Andrea Montanari. Discussing a problem with Andrea has been always a humbling but eye-opening experience. His modesty has given me the courage to knock the door of his office whenever I need his expertise. Thanks Andrea!

I would also like to thank my associate advisor Professor Robert M. Gray. I doubt that anyone who interacts with Professor Gray can fail to be impressed by his tremendous intellectual energy. His big pictures and his vast knowledge of the literature in information theory, compression, image, and audio processing has always helped me in choosing research directions. I would also like to thank Professor Gray for his valuable comments and feedback on this thesis.

My appreciation also goes to Professor Iain Johnstone for both chairing on my committee and teaching me statistics. I have taken two courses, ‘Theory of statistics’ and ‘Function estimation in noise’, with Iain and they were two of the best-taught courses I have taken at Stanford.

I wish to express my appreciation to Professor Bala Rajaratnam. Bala’s high motivation and energy for doing research and exploring new topics have been always inspiring to me. I have also benefited from his vast knowledge in statistics.

Heather, Helen, Angie, and Nora have always been there to help me with the administrative tasks. I am grateful to them.

I was very fortunate to have many good friends around me at Stanford: Morteza, Nima, Rostam, Navid, Amir, Bitan, Vahbod, Shadi, Ali, Maryam, Rahul, Satish, Behrad, Neda, Maryam, Ian, Kjartan, Marie, Mortan, Yaniv, Sewoong, Mohammad, Tahereh, Paul, Sam, Shahriar, Hossein, Mohsen. Thank you all!

I would also like to take this opportunity to thank my parents for their unconditional love and support. They have always believed in me, more than I myself do and have been fully supportive of all my decisions. I would like to dedicate this thesis to them for their endless love, support, and self-sacrifices.

If it was not for the experiences my sister Atoosa and my brother Arash passed on to me, I could not be here. I would like to thank them for their help, love, and support. Finally my deepest gratitude goes to my wife, Shirin. She is not only the love of my life, but also my favorite classmate, my best friend, and my best collaborator. Every time I have an idea, Shirin is the first person who reviews it and gives valuable feedback and suggestions. Without her endless encouragement and indispensable support, I could never be even close to the end of my PhD journey. I hope I can be as encouraging and as supportive for her as she has been for me. Stanford University will always be one of my most favorite places in the world not only because I spent several years here, but also because every place here reminds me of a good memory Shirin and I have had together. I hope the nonstop construction work at Stanford leaves some of these places untouched so that we can come back and refresh our sweet memories in the future.

Contents

Preface	v
Notational Conventions	vii
Acknowledgements	ix
1 Introduction	1
1.1 Shannon-Nyquist-Whittaker sampling	1
1.2 Compressed sensing	2
1.3 Applications of compressed sensing	5
1.3.1 Magnetic resonance imaging	6
1.3.2 Imaging	6
1.3.3 Seismic data collection	7
1.3.4 Other applications	9
1.4 Organization and contribution of the thesis	9
1.4.1 Chapter 2	9
1.4.2 Chapter 3	9
1.4.3 Chapter 4	10
1.4.4 Chapter 5	10
1.4.5 Chapter 6	11
1.4.6 Chapter 7	11
1.4.7 Chapter 8	11

2	Deterministic Results	12
2.1	ℓ_1 -minimization	13
2.2	Orthogonal matching pursuit (OMP)	14
2.3	Iterative hard thresholding (IHT)	16
2.4	Iterative soft thresholding (IST)	18
2.5	Connection between RIP and coherence	19
2.6	Implications for compressed sensing	20
	2.6.1 Coherence of random measurement matrices	20
	2.6.2 Analysis of RIP for random matrices	25
2.7	Other algorithms	30
3	Maximin Framework	31
3.1	Phase transitions	32
	3.1.1 Maximin tuning	34
3.2	Iterative algorithms	36
	3.2.1 Simple iterative algorithms	36
	3.2.2 Composite iterative algorithms	37
	3.2.3 Threshold choice	38
3.3	Estimating the empirical phase transition	39
3.4	Tuning procedure	40
3.5	Tuning results	41
3.6	Recommended choices	41
3.7	Robustness	46
3.8	Running times	50
3.9	Ensembles based on fast operators	51
3.10	Before using these results	52
3.11	The computational effort-phase transition tradeoff	54
4	Approximate Message Passing	56
4.1	Introduction	56
	4.1.1 Underdetermined linear systems	58
	4.1.2 Phase transitions	59

4.2	Statistical heuristic for iterative approaches	59
4.2.1	State evolution	62
4.3	Message passing algorithm	64
4.3.1	Optimizing the MP phase transition	68
4.4	Other thresholding policies	70
4.5	Universality	76
4.6	Simulation results	76
4.6.1	Data generation	76
4.6.2	Estimating phase transitions	77
4.6.3	Tuning of algorithms	79
4.6.4	Empirical phase transition	79
4.6.5	Example of the interference heuristic	80
4.6.6	Testing predictions of state evolution	80
4.6.7	Coefficient universality	82
4.6.8	Matrix universality	82
4.6.9	Timing results	83
5	Designing Approximate Message Passing	96
5.1	Contribution and organization	97
5.1.1	Notation	97
5.1.2	Basis pursuit problem	98
5.1.3	BPDN problem	99
5.1.4	Theoretical prediction	100
5.1.5	Extensions	101
5.1.6	Comparison with other work	101
5.1.7	Organization	103
5.2	AMP for hard constraints	103
5.2.1	Construction of the graphical model	104
5.2.2	Large system limit	105
5.2.3	Large β limit	107
5.2.4	From message passing to AMP	108

5.3	AMP for soft constraints	111
5.3.1	Construction of the graphical model	111
5.3.2	Large system limit	112
5.3.3	Large β limit	114
5.3.4	From message passing to AMP	115
5.3.5	Comments	115
5.4	State evolution	116
5.4.1	Exactly sparse solution	117
5.5	Extensions	122
5.5.1	Positivity constraint	122
5.5.2	AMP for reconstruction with prior information	126
5.6	Advantages	129
6	Noise Sensitivity Phase Transition	131
6.1	Introduction	131
6.1.1	Results of our formalism	133
6.2	Minimax MSE of soft thresholding	135
6.3	Main results	139
6.3.1	Terminology	139
6.3.2	Results of the formalism	141
6.3.3	Interpretation of the predictions	143
6.3.4	Comparison to other phase transitions	144
6.3.5	Validating the predictions	145
6.4	The formalism	145
6.4.1	The AMPT algorithm	145
6.4.2	Formal MSE, and its evolution	146
6.4.3	AMPT-LASSO calibration	154
6.4.4	Derivation of proposition 6.3.7	156
6.4.5	Formal MSE above phase transition	159
6.5	Empirical validation	161
6.5.1	Below phase transition	161

6.5.2	Above phase transition	171
6.6	Extensions	171
6.6.1	Positivity constraints	171
6.6.2	Other classes of matrices	174
6.7	Tables	174
7	AMP Versus Other First Order Methods	180
7.1	Proximity operator	181
7.1.1	Unconstrained problem	181
7.1.2	Optimization over ‘easy’ convex sets	182
7.1.3	‘Easy’ non-differentiable functions	183
7.1.4	One framework for all	185
7.1.5	First order methods for compressed sensing	187
7.2	Analyzing the convergence rate	188
7.2.1	Drawback of deterministic approach	188
7.2.2	Linear system: motivating example	190
7.2.3	Average case convergence for CS problems	194
7.3	Difficulty of compressed sensing problems	194
7.3.1	Notations	195
7.3.2	Matrix universality	195
7.3.3	Coefficient ensemble	196
7.3.4	Sparsity level	200
7.4	Other first-order methods	202
7.4.1	Regularization parameter and continuation strategies	203
7.4.2	Other Heuristics	204
7.4.3	TwIST	205
7.4.4	GPSR-BB	205
7.4.5	FPC-AS	206
7.4.6	Properties of statistical convergence rates	206
7.5	Comparison results	207
7.5.1	Exact recovery problem	207

7.5.2	Measurement noise	209
7.6	Detailed experiments and proofs	209
7.6.1	AMPT-LASSO calibration	209
7.6.2	Discussion of matrix universality	211
7.6.3	Discussion of coefficient distribution experiments	215
7.6.4	Discussions of the sparsity level	221
8	Future Directions	232
8.1	Other sparsity patterns	232
8.2	Correctness of state evolution	234
8.3	Convergence rate of first-order methods	235
A	Sub-gaussian and Sub-exponential Random Variables	236
B	Berry-Esseen Central Limit Theorem	240
C	Wigner’s Semicircle Law	242
D	Proofs of Chapter 2	244
D.1	Proof of Theorem 2.3.1	244
D.2	Proof of Theorem 2.4.1	249
E	Proofs of Chapter 4	254
E.1	proof of Lemma 4.2.1	254
E.2	Proof of Proposition 4.2.2	256
E.2.1	proof for the case $\chi = +$	257
E.2.2	Proof of Lemma 5.2.4	261
E.2.3	Proof of Theorem 4.3.1	262
E.2.4	Proof of Theorem 4.4.2	262
F	Proofs of Chapter 7	265
F.1	Proof of Theorem 7.2.1	265
	Bibliography	267

List of Tables

3.1	Recommended IST: Optimally tuned FAR parameter and resulting optimized phase transition, both as a function of δ . Recommended value of relaxation parameter $\kappa = 0.6$	44
3.2	Recommended IHT: Optimally tuned FAR parameter and resulting optimized phase transition, both as a function of δ . Recommended value of relaxation parameter $\kappa = 0.65$	44
3.3	Recommended TST: optimal tuning parameters are $\alpha = \beta = 1$	44
3.4	Algorithm Timings at the Standard Suite. Average running time (sec) until $\frac{\ y - A\hat{x}\ _2}{\ y\ _2} \leq .001$. Averages cover 10 independent runs. Problem sizes as indicated.	50
3.5	Recommended IST, Standard FastOps Suite. Optimally Tuned FAR and optimized phase transition ρ^* of the recommended IST algorithm. Recommended relaxation parameter $\kappa = 1$	53
3.6	Recommended IHT, Standard FastOps Suite. Optimally Tuned FAR and optimized phase transition ρ^* of the recommended IHT algorithm. Recommended relaxation parameter $\kappa = 1$	53
3.7	Algorithm Timings at the Standard FastOps Suite. Average running time (sec) until $\frac{\ y - A\hat{x}\ _2}{\ y\ _2} \leq .001$. Averages cover 10 independent runs. Problem sizes as indicated.	54
4.1	Timing Comparison of AMP and LARS. Average Times in CPU seconds.	84
6.1	Some observables and their names.	152

6.2	Parameters of quasi-least-favorable settings studied in the empirical results presented here.	162
6.3	Results at $N = 1500$. MSE of $\text{LASSO}(\lambda^*)$ at nearly-least-favorable situations, together with standard errors (SE)	163
6.4	Results at $N = 4000$. Theoretical and empirical MSE's of $\text{LASSO}(\lambda^*)$ at nearly-least-favorable situations, together with standard errors (SE). 164	
6.5	Results at $N = 8000$. Theoretical and empirical MSE's of $\text{LASSO}(\lambda^*)$ at nearly-least-favorable situations with $\delta = 0.10$, together with standard errors (SE) of the empirical MSE's	165
6.6	Results above Phase transition. Parameters of the construction as well as theoretical predictions and resulting empirical MSE figures	175
6.7	$N = 1500$, λ dependence of the MSE at fixed μ	176
6.8	$N = 4000$, λ dependence of the MSE at fixed μ	177
6.9	$N = 1500$, μ dependence of the MSE at fixed λ	178
6.10	$N = 1500$, MSE for 5-point prior	179
7.1	Matrix ensembles considered in the matrix universality hypothesis tests. 196	
7.2	Coefficient ensembles considered in coefficient ensemble analysis experiments.	198
7.3	Number of iterations of each algorithm in matrix universality experiments.	211
7.4	Performance of different algorithms on several compressed sensing problem instances. Here matrix ensemble is $N(0, 1/n)$ and the coefficient ensemble is 3P. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.	229
7.5	Performance of different algorithms on several compressed sensing problem instances. Here the matrix ensemble is $N(0, 1/n)$ and the coefficient ensemble is Cauchy. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.	230

7.6	Performance of different algorithms on several compressed sensing problem instances. Here the matrix ensemble is partial-DCT and the coefficient ensemble is $3P$. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.	230
7.7	Performance of different algorithms on several compressed sensing problem instances in the presence of measurement noise. Here the matrix ensemble is $N(0, 1/n)$ and coefficient ensemble is $3P$. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.	231

List of Figures

1.1	Left: Original Barbara image. Right: Number of coefficients in intervals of size 10. The number of coefficients is shown in the logarithmic scale.	3
1.2	Left: Original Barbara image. Right: Barbara image reconstructed from ten percent of the wavelet coefficients.	3
1.3	Picture of an MRI machine	6
1.4	Rice University single pixel camera.	7
1.5	Random Lens imaging device.	8
1.6	Seismic data collection procedure.	8
2.1	Geometric intuition of iterative hard thresholding algorithm. At every iteration the algorithm moves toward the hyperplane $y = Ax$ and then promotes sparsity by the hard thresholding function.	17
3.1	<i>Phase Diagram for ℓ_1 minimization.</i> horizontal axis: indeterminacy $\delta = n/N$. vertical axis: sparsity $\rho = k/n$. Shaded attribute depicts limiting probability that ℓ_1 successfully reconstructs the sparsest solution, as a function of the state variables (ρ, δ) . Dark region: limiting probability 0. Light region: limiting probability is 1.	33

3.2	<i>Finite-N Phase Transitions.</i> Fraction of unsuccessful recovery attempts by IHT, as a function of sparsity level ρ . Here $\delta = .5$ and $\rho = k/n$ is varying. Results are shown at 3 values of N : 500, 1000, 4000; note steepening of the transition as N increases, in accord with theory. At each unique parameter combination, twenty random problem instances were tried.	35
3.3	(a) Optimized phase transitions for IST at various choices of relaxation parameter κ . (b) FAR parameter choices yielding those optima. The value $\kappa = 0.6$ outperforms the other choices.	42
3.4	(a) Optimized phase transitions for IHT at various choices of relaxation parameter κ . (b) FAR parameter choices yielding those optima. The value $\kappa = 0.6$ outperforms the other choices.	42
3.5	(a) Empirical phase transitions of TST- (α, β) for $\alpha = 1$ and different values of β ; $\beta = 1$ performs best; (b) Empirical phase transitions when $\alpha = \beta$; again $\beta = 1$ performs best. In both panels, differences are small for $\delta < .5$, i.e. undersampling factors greater than 2.	43
3.6	Phase Transitions of several algorithms at the standard suite. Upper curve: theoretical phase transition, ℓ_1 minimization; lower curves: Observed transitions of algorithms recommended here.	45
3.7	Observed phase transitions of recommended IST at different coefficient ensembles.	47
3.8	Observed phase transition of recommended IHT at different coefficient ensembles.	47
3.9	Observed phase transition of recommended TST at different coefficient ensembles.	48
3.10	Observed phase transition of recommended IST for different matrix ensembles.	48

3.11	Observed phase transition of recommended IHT for different matrix ensembles. Note that the red curve (URP matrix ensemble) is significantly below the blue curve (USE matrix ensemble) for $0.3 < \delta < .85$. This was the only substantial exception observed to the maximin property in our study.	49
3.12	Observed phase transition of recommended TST for different matrix ensembles.	49
3.13	Comparison of the performance of recommended IHT, IST and TST for partial fourier ensemble.	52
3.14	(a) Phase transitions of recommended IHT for different matrix ensembles (b) Phase transitions of recommended IST for different matrix ensembles. Generally speaking, the partial Fourier ensemble is not the worst case ensemble; note especially the case <i>IST</i> with $\delta > 0.4$	55
4.1	The phase transition lines for reconstructing sparse non-negative vectors (problem +, red), sparse signed vectors (problem \pm , blue) and vectors with entries in $[-1, 1]$ (problem \square , green). Continuous lines refer to analytical predictions from combinatorial geometry or the state evolution formalisms. Dashed lines present data from experiments with the AMP algorithm, with signal length $N = 1000$ and $T = 1000$ iterations. For each value of δ , we considered a grid of ρ values, at each value, generating 50 random problems. The dashed line presents the estimated 50th percentile of the response curve. At that percentile, the root mean square error after T iterations obeys $\sigma_T \leq 10^{-3}$ in half of the simulated reconstructions.	60

4.2 Development of fixed points for formal MSE evolution. Here we plot $\Psi(\sigma^2) - \sigma^2$ where $\Psi(\cdot)$ is the MSE map for $\chi = +$ (left column), $\chi = \pm$ (center column) and $\chi = \square$ (right column), $\delta = 0.1$ (upper row, $\chi \in \{+, \pm\}$), $\delta = 0.55$ (upper row, $\chi = \square$), $\delta = 0.4$ (lower row, $\chi \in \{+, \pm\}$) and $\delta = 0.75$ (lower row, $\chi = \square$). A crossing of the y-axis corresponds to a fixed point of Ψ . If the graphed quantity is negative for positive σ^2 , Ψ has no fixed points for $\sigma > 0$. Different curves correspond to different values of ρ : where ρ is respectively less than, equal to and greater than ρ_{SE} . In each case, Ψ has a stable fixed fixed point at zero for $\rho < \rho_{SE}$, and no other fixed points, an unstable fixed point at zero for $\rho = \rho_{SE}$ and develops two fixed points at $\rho > \rho_{SE}$. Blue curves correspond to $\rho = \rho_{SE}(\delta; \chi)$, green to $\rho = 1.05 \cdot \rho_{SE}(\delta; \chi)$, red to $\rho = 0.95 \cdot \rho_{SE}(\delta; \chi)$ 63

4.3 Observed phase transitions of reconstruction algorithms. Algorithms studied include iterative soft and hard thresholding, orthogonal matching pursuit, and related. Parameters of each algorithm are tuned to achieve the best possible phase transition [76]. Reconstructions signal length $N = 1000$. Iterative thresholding algorithms used $T = 1000$ iterations. Phase transition curve displays the value of $\rho = k/n$ at which success rate is 50%. 67

4.4 Observed Phase Transitions for 6 Algorithms, and ρ_{SE} . AMP: method introduced in main text. IST: Iterative Soft Thresholding. IHT: Iterative Hard Thresholding. TST: a class of two-stage thresholding algorithms including subspace pursuit and CoSaMP. OMP: Orthogonal Matching Pursuit. Note that the ℓ_1 curve coincides with the state evolution transition ρ_{SE} , a theoretical calculation. The other curves show empirical results. 86

4.5	QQPlots tracking marginal distribution of mutual access interference (MAI). Panels (a)-(i): iterations 10, 20, . . . , 90. Each panel shows QQ-plot of MAI values versus normal distribution in blue, and in red (mostly obscured) points along a straight line. Approximate linearity indicates approximate normality. Decreasing slope with increasing iteration number indicates decreasing standard deviation as iterations progress.	87
4.6	QQPlots tracking marginal distribution of mutual access interference (MAI). Matrix Ensemble: partial Fourier. Panels (a)-(i): iterations 30, 60, . . . , 270. For other details, see Figure 4.5.	88
4.7	Comparison of State Evolution predictions against observations. $\rho = .15$, $\delta = .3$. Panels (a)-(d): MSENZ, MSE, MDR, FAR. Curve in red: theoretical prediction. Curve in blue: mean observable. Each panel shows the evolution of a specific observable as iterations progress. Two curves are present in each panel, however, except for the lower left panel, the blue curve (empirical data) is obscured by the presence of the red curve. The two curves are in close agreement in all panels.	89
4.8	Comparison of State Evolution predictions against observations. $\rho = 0.2$, $\delta = 0.5$. For details, see Figure 4.7.	90
4.9	Comparison of State Evolution predictions against observations for $\rho = 0.36$, $\delta = 0.7$. For details, see Figure 4.7.	91
4.10	Comparison of Failure probabilities for different ensembles. In the left window, $\delta = 0.10$ and in the right window $\delta = 0.3$. Red: unit-amplitude coefficients. Blue: uniform $[-1, 1]$. Green: Gaussian. Black: Cauchy. Points: observed success fractions Curves: Logistic fit.	92
4.11	Observed Phase Transitions at different matrix ensembles. Case $\chi = \pm$. Red: Uniform Spherical Ensemble (Gaussian with normalize column lengths). Magenta: Rademacher (± 1 equiprobable). Green: partial Fourier. Blue: ρ_{ℓ_1}	93

4.12	Iteration Counts versus Signal Length N . Different curves show results for different stopping rules. Horizontal axis: signal length N . Vertical axis: Number of iterations, T . Blue, Green, Red, Aqua curves depict results when stopping thresholds are set at $2^{-13-\ell}$, with $\ell = 0, 1, 2, 3$. Each doubling of accuracy costs about 5 iterations.	94
4.13	CPU Time Scaling with N . Different curves show results for different stopping rules. Horizontal axis: signal length N . Vertical axis: CPU time(seconds). Blue, Green, Red, Aqua curves depict results when stopping thresholds are set at $2^{-13-\ell}$, with $\ell = 0, 1, 2, 3$	95
5.1	Comparison of state evolution predictions against observations. Top row: $N = 4000$, $\delta = .2$ and $\rho = 0.05$. Bottom row: $N = 4000$, $\delta = .3$ and $\rho = 0.17$. Each red point is the average of 50 Monte Carlo samples and the bars show the 95 percent confidence interval.	118
5.2	The maximin optimal values of θ as proposed in [38].	121
5.3	Theoretical phase transition of AMPA after (a)50 (b) 100 (c) 200 and (d) 500 iterations. Dotted line is the phase transition curve of the basis pursuit problem derived in [44] and [38].	123
6.1	Contour lines of the minimax noise sensitivity $M^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted black curve graphs the phase boundary $(\delta, \rho_{\text{MSE}}(\delta))$. Above this curve, $M^*(\delta, \rho) = \infty$. The colored lines present level sets of $M^*(\delta, \rho) = 1/8, 1/4, 1/2, 1, 2, 4$ (from bottom to top).	134
6.2	Left: $M^\pm(\epsilon)$ as a function of ϵ ; Right: $\tau^\pm(\epsilon)$ as a function of ϵ	137
6.3	Illustration of α -least-favorable ν . For $\epsilon = 1/10$, we consider soft thresholding with the minimax parameter $\tau^\pm(\epsilon)$. We identify the smallest μ such that the measure $\nu_{\epsilon, \mu} = (1 - \epsilon)\delta_0 + \frac{\epsilon}{2}\delta_\mu + \frac{\epsilon}{2}\delta_{-\mu}$ has $\text{mse}(\nu_{\epsilon, \mu}, \tau^*) \geq 0.98 M^\pm(0.1)$ (i.e. the MSE is at least 98% of the minimax MSE).	138

- 6.4 Contour lines of the near-least-favorable signal amplitude $\mu^*(\delta, \rho, \alpha)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while the colored solid lines portray level sets of $\mu^*(\delta, \rho, \alpha)$. The 3-point mixture distribution $(1 - \epsilon)\delta_0 + \frac{\epsilon}{2}\delta_\mu + \frac{\epsilon}{2}\delta_{-\mu}$, ($\epsilon = \delta\rho$) will cause 98% of the worst-case MSE. When a k -sparse vector is drawn from this distribution, its nonzeros are all at $\pm\mu$ 143
- 6.5 Contour lines of the maximin penalization parameter: $\lambda^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while thin lines are contours for $\lambda^*(\delta, \rho, \alpha)$. Close to phase transition, the maximin value approaches 0. 144
- 6.6 MSE Map Ψ in three cases, and associated fixed points. Left: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}/2$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$ Center: $\delta = 0.25$, $\rho = \rho_{\text{MSE}} \times 0.95$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$ Right: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$ 148
- 6.7 Crossing the phase transition: effects on MSE Map Ψ , and associated state evolution. Left: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}/2$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$ Middle: $\delta = 0.25$, $\rho = 0.9 \cdot \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$ Right: $\delta = 0.25$, $\rho = 1.5 \cdot \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$. In each case $\tau = \tau^\pm(\delta\rho)$. 151
- 6.8 Contour lines of $\tau^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while thin lines are contours for $\tau^*(\delta, \rho)$. 157
- 6.9 Finite- N scaling of empirical MSE. Empirical MSE results from the cases $N = 1500$, $N = 4000$ and $N = 8000$ and $\delta = 0.1$. Vertical axis: empirical MSE. Horizontal axis: $1/N$. Different colors/symbols indicate different values of the sparsity control parameter δ . Vertical bars denote $\pm 2SE$ limits. Theoretical predictions for the $N = \infty$ case appear at $1/N = 0$. Lines connect the cases $N = 1500$ and $N = \infty$. . 166
- 6.10 Saddlepoint in formal MSE. Right panel: Behavior of formal MSE as λ is varied away from λ^* . Left panel: Behavior of formal MSE as μ is varied away from μ^* in the direction of smaller values. Black lines indicate locations of μ^* and λ^* . $\delta = 0.25$, $\rho = \rho_{\text{MSE}}(\delta)/2$ 167

6.11	Scatterplots comparing Theoretical and Empirical MSE's found in Tables 6.7 and 6.8. Left Panel: results at $N = 1500$. Right Panel: results at $N = 4000$. Note visible tightening of the scatter around the identity line as N increases.	168
6.12	Convexity structures in formal MSE. Behavior of formal MSE of 5 point mixture combining nearly least-favorable μ with discount of 1% and one with discount of 50%. Also, the convexity bound (6.25) and the formal MSE of associated 3-point mixtures is displayed. $\delta = 0.25$, $\rho = \rho_{\text{MSE}}(\delta)/2$	170
6.13	Contour lines of the positivity-constrained minimax noise sensitivity $M^{*,+}(\delta, \rho)$ in the (ρ, δ) plane. The dotted black curve graphs the phase boundary $(\delta, \rho_{\text{MSE}}^+(\delta))$. Above this curve, $M^{*,+}(\delta, \rho) = \infty$. The colored lines present level sets of $M^{*,+}(\delta, \rho) = 1/8, 1/4, 1/2, 1, 2, 4$ (from bottom to top). . . .	173
7.1	Comparison of the theoretical asymptotic prediction provided in Example 7.1 for a linear system and the Monte Carlo simulation result on a medium size problem, $n = 1000$. Elements of x_o are iid $N(0, 1)$, 1000 Monte Carlo simulations are considered. The bars show 95 percent confidence intervals.	193
7.2	Checking the matrix universality hypothesis. Left: logarithm of the mean square error of FISTA for four different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .001$. Right: logarithm of the mean square error of IST for four different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .1$. The four curves shown in the above figures are on top of each other and this confirms the matrix universality hypothesis. For the statistical analysis of the discrepancies refer to Section 7.6.2.	197

7.3	<p>Logarithm of the relative mean square error for several distributions. 3P stands for three point prior or constant amplitude prior. The other abbreviations are explained in table 7.2. Left: $\delta = .5$, $\epsilon = .15$, $\lambda = .01$, $\sqrt{\nu} = .001$ and the algorithm is FISTA. Right: $\delta = .3$, $\epsilon = .03$, $\lambda = 0.01$, $\sqrt{\nu} = 0.01$ and the algorithm is IST. These simulations confirm the fact that 3P is less favorable than the other distributions.</p>	200
7.4	<p>The values of λ that lead to the same relative mean square error. The elements of the measurement matrix are iid $N(0, 1/n)$ and the coefficients are drawn from 3-point prior $(1 - \epsilon)\delta_0 + \epsilon/2\delta_1 + \epsilon/2\delta_{-1}$. The standard deviation of the measurement noise is $\sigma = .001$. The dashed curve represents $.95\rho_{SE}$. The results are derived from the theoretical calculations.</p>	202
7.5	<p>Comparison of convergence of FISTA and IST algorithms for different values of ρ. Left: $\delta = .05$, $N = 1500$, $\sigma = .001$ and the algorithm is FISTA. Right: $\delta = 0.32$, $N = 1500$, $\sigma = .001$ and the algorithm is IST. These simulation confirm the hypothesis that larger values of ϵ are less favorable.</p>	203
7.6	<p>Performance of FISTA in terms of λ. Left: $\delta = .3$, $\sqrt{\nu} = .01$, $N = 2000$, $\rho = .5\rho_{Se}$; Right: the same problem with $\nu = 0$</p>	204
7.7	<p>Checking matrix universality hypothesis. Top-Left: logarithm of the mean square error of FISTA for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .001$. Bottom-Left: The p-values of the test specified in (7.18). Top-Right: logarithm of the mean square error of IST for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .1$. Bottom-right: The p-values of the test specified in (7.18). Clearly the p-values are large for both experiments and therefore we cannot reject the null hypothesis.</p>	213

7.8	Checking matrix universality hypothesis of FPC-AS. Top-left: logarithm of the mean square error for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .75\rho_{SE}$, $\lambda = .001$, $\nu = 0$. Bottom-left: The p-values of the test specified in (7.18). Top-right: logarithm of the mean square error for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .001$, $\nu = 0$. Bottom-right: The p-values of the test specified in (7.18). Clearly the p-values are large for both experiments and therefore we cannot reject the null hypothesis.	214
7.9	Matrix universality results for Left: AMPT with $\delta = .5$, $\rho = .5\rho_{SE}$, $\tau = 2$, $\nu = 0$. Right: TWIST with $\delta = .5$, $\rho = .5\rho_{SE}$, $\lambda = .001$, $\nu = 0$. In both simulations the number of Monte Carlo samples is 80.	215
7.10	Performance of AMPT algorithm on a few different distributions. In this simulation $N = 4000$, $\delta = .5$, $\epsilon = .075$, $\tau = 2$, $\sqrt{\nu} = .001$ and the number of Monte Carlo samples 25. For more information on the decrease of the discrepancy between theory and Monte Carlo simulation refer to [41].	219
7.11	Comparison of $t(\alpha)$ for two different coefficient distributions specified in Table 7.2. Top-left: $t(\alpha)$ for FISTA, at $\delta = .3$, $\epsilon = .03$, $\lambda = .001$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Top-right: $t(\alpha)$ for IST, $\delta = .3$, $\epsilon = .07$, $\lambda = .05$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Large p-values indicate that we cannot reject the null.	221
7.12	Comparison of $t(\alpha)$ for two different distributions on Top-left: $t(\alpha)$ for TWIST at $\delta = .3$, $\epsilon = .07$, $\lambda = .05$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Top-right: $t(\alpha)$ of FPCAS at $\delta = .3$, $\epsilon = .07$, $\lambda = 0.01$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Large p-values indicate that we cannot reject the null.	222
7.13	Comparison of different sparsity levels on AMPT algorithm $\delta = .5$, $\epsilon = .04$, $N = 4000$, $\tau = 1.8$, $\sqrt{\nu} = .001$, and the coefficient distribution is 3P. The empirical results are the result of 100 Monte Carlo samples.	226

7.14 Comparison of different values of ρ or ϵ for TWIST and FPCAS. Left: $\delta = .07$, $N = 1500$, $\sigma = .001$, Algorithm = TWIST, matrix ensemble = $N(0, 1/n)$ and Coefficient ensemble is $F_{\epsilon,1}$. Right: $\delta = .48$, $N = 1500$, $\sigma = .001$, Algorithm = FPC-AS, matrix ensemble = $N(0, 1/n)$ and Coefficient ensemble is $F_{\epsilon,1}$. The sudden drops in FPC-AS correspond to the steps in which is solving a linear system. 228

Chapter 1

Introduction

1.1 Shannon-Nyquist-Whittaker sampling

One of the most fundamental building blocks in digital signal processing systems is the analog to digital converter. This block converts an analog signal to bits that can then be used by digital systems to serve different purposes such as transmission, storage, estimation, etcetera. The first step of analog to digital conversion is to discretize the signal in time, which is called sampling. The common practice in acquisition of a signal is to sample it according to the celebrated Shannon-Nyquist-Whittaker theorem and reconstruct it with linear interpolation techniques. Consider an analog signal $f(t)$ with frequency domain representation $F(j\Omega)$. The following theorem which is due to Nyquist and Whittaker [110], [85] and is popularized by Shannon [95], shows how bandlimited signals can be sampled without losing any information.

Theorem 1.1.1. *If $h_T(t) = \sin(\pi t/T)/(\pi t/T)$, then $\{h_T(t-nT)\}_{n \in \mathbb{Z}}$ is an orthogonal basis for the space U_T of functions whose Fourier transform has a support included in $[-\pi/T, \pi/T]$. If $f \in U_T$ then,*

$$f(nT) = \frac{1}{T} \langle f(t), h_T(t - nT) \rangle. \quad (1.1)$$

According to this theorem if $f \in U_T$, then

$$f(t) = \lim_{n \rightarrow \infty} \sum_{-n}^n f(nT) h_T(t - nT), \quad (1.2)$$

where the convergence is in the L_2 -norm sense. In other words, if we just keep the samples of the original signal at times nT , we will still be able to reconstruct the analog signal and the reconstruction is given by (1.2). Furthermore, it is not difficult to see that if we sample the signal at a lower rate, there are some signals in this class that can not be reconstructed correctly. Therefore, the sampling rate suggested by Theorem 1.1.1 seems to be optimal in the sense that sampling at a lower rate may lead to a loss of information.

However, researchers in image, video, and audio processing have observed that the Shannon-Nyquist-Whittaker sampling is not ‘efficient’ or ‘optimal’. To understand this claim, consider the Barbara image shown in Figure 1.1. As exhibited in this figure, the wavelet representation of the image is ‘approximately sparse’, i.e., most of the coefficients are either zero or close to zero. The fast decay in the histogram of the wavelet coefficients confirms this phenomenon. As is shown in Figure 1.2, if we knew the locations of the large wavelet coefficients, we could have used the approximate sparsity structure present in the images and just measured the large coefficients. However, in many applications the locations of the large coefficients are not known beforehand. Therefore, there is a need for a new sampling paradigm that demands fewer measurement of a sparse or nearly sparse signal without any knowledge of the location of “important coefficients”. Compressed sensing [20], [31] is a field of research that addresses this issue. In the next section, after the formal explanation of our assumptions on the signal spaces, we discuss the basic ideas of compressed sensing.

1.2 Compressed sensing

Compressed sensing exploits the ‘sparsity’ to sample the signal more efficiently than the classic Shannon-Nyquist scheme. Suppose the vector $x_o \in \mathbb{R}^N$, to be acquired,

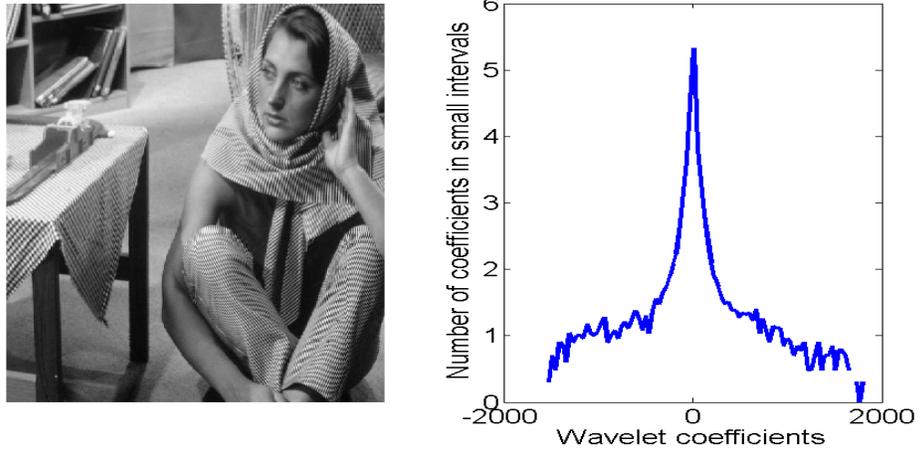


Figure 1.1: Left: Original Barbara image. Right: Number of coefficients in intervals of size 10. The number of coefficients is shown in the logarithmic scale.



Figure 1.2: Left: Original Barbara image. Right: Barbara image reconstructed from ten percent of the wavelet coefficients.

has an additional structure and that is, there exists a basis of \mathbb{R}^N , called G , such that $\theta = Gx_o$ is sparse. There are several different notions of sparsity defined in the compressed sensing literature. An important example is the class of ℓ_p -balls. In these classes it is assumed that $\theta \in B_p^N(C) = \{\theta \in \mathbb{R}^N \mid \|\theta\|_p \leq C\}$ for $p \in [0, 1]$ and a constant C .¹ Here $\|\theta\|_p = (\sum |\theta_i|^p)^{1/p}$ for $p > 0$. It is straightforward to confirm that if $\theta \in B_p^N(C)$ and $\theta_{(k)}$ represents the k^{th} largest coefficient of θ , then

$$|\theta_{(k)}| \leq \frac{C}{k^{\frac{1}{p}}}. \quad (1.3)$$

Therefore, smaller values of p correspond to faster decay in the magnitude of the coefficients. The case $p = 0$ is of special interest. The ℓ_0 -norm is defined as

$$\|\theta\|_0 = |\{i : \theta_i \neq 0\}|, \quad (1.4)$$

i.e., the number of non-zero elements of x_i . In this case the signal is called exactly sparse since it has at most $\lfloor C \rfloor$ non-zero coefficients. In this thesis we consider the ℓ_0 -balls. However, the extensions to the more general class of ℓ_p -balls will be discussed briefly in Chapter 8.

So far we have explained the first ingredient of compressed sensing which is the structure of the signals. The next important ingredient is the “measurement operator”. We should keep in mind that the goal is to reduce the number of measurements. The simplest way of making these measurements is through a linear operator, i.e., the measurements are given by $y = Ax_o$, where A is an $n \times N$ measurement matrix. Noise may also be present in the measurement process and therefore we will also consider a more general model $y = Ax_o + z$, where z is the measurement noise. We will discuss the statistics of the noise later in the thesis. Clearly, the assumption is that the number of measurements is less than the dimension of x_o ($n < N$). More interestingly, randomness is used for generating the measurement matrix A [31], [20]. For example, A can be a matrix whose elements are independently drawn from Gaussian, Bernoulli,

¹Although we use the word norm for $\|\cdot\|_p$, it is not an actual norm when $p \in [0, 1)$ since it does not satisfy the triangle inequality.

or any other sub-gaussian distribution. Refer to Appendix A for more information on sub-gaussian matrices. Since $n < N$, the system of equation $y = Ax$ is not invertible. Hence, the structure of the signal must be incorporated in the recovery algorithm or otherwise the recovery will not be successful. Hence, the compressed sensing recovery algorithms are more sophisticated than the Shannon-Nyquist sinc interpolation. This issue will be discussed in detail in Chapters 2, 3, 4.

In the rest of the thesis for notational simplicity we assume that $G = I$; the sparsity is happening in the time domain. To understand the intuition of the sparse recovery algorithms, suppose that the original signal x_o is exactly sparse. Since $y = Ax_o$ is an undersampled system of linear equation, it has infinitely many solutions. Among these solutions we are interested in the sparsest one. This can be done by the following minimization algorithm:

$$\begin{aligned} \min \|x\|_0, \\ \text{s.t. } y = Ax. \end{aligned} \tag{1.5}$$

If the sparsest solution is unique in the sense that there is no other solution at the same sparsity level, (1.5) will recover that solution. Although (1.5) is the most natural recovery algorithm, it can not be used in practice since it is an NP-complete problem. Many different heuristic approaches have been proposed with different performance guarantees and different computational complexities. In the next chapter we discuss some of these approaches and derive their deterministic performance guarantees. Before finishing up this chapter let us summarize some of the applications of compressed sensing and the contributions of this thesis.

1.3 Applications of compressed sensing

Although compressed sensing is a new field, it has already found many applications. In this section we briefly review some of these applications.



Figure 1.3: Picture of an MRI machine

1.3.1 Magnetic resonance imaging

One of the interesting applications of compressed sensing is in magnetic resonance imaging or MRI. Figure 1.3 shows a picture of an MRI machine. In order to take an image of a tissue, organ, or joint, a person should stay still inside the machine for 10-45 minutes depending on the resolution needed. This may be difficult specially for children. The best way to reduce the time of acquisition is to reduce the number of samples. As mentioned before in Shannon-Nyquist framework, decreasing the number of samples is equivalent to aliasing artifacts in the reconstructed images. However, CS can reduce the number of measurements by exploiting the sparsity of MRI images. The reduction in the acquisition time may help the real time imaging as well. This is extremely useful in some application such as functional MRI. For more information on compressed sensing MRI refer to [73] and the references therein.

1.3.2 Imaging

In some types of camera the sensors are either large or expensive. As a very simple example consider SLR cameras that are usually larger and more expensive than the

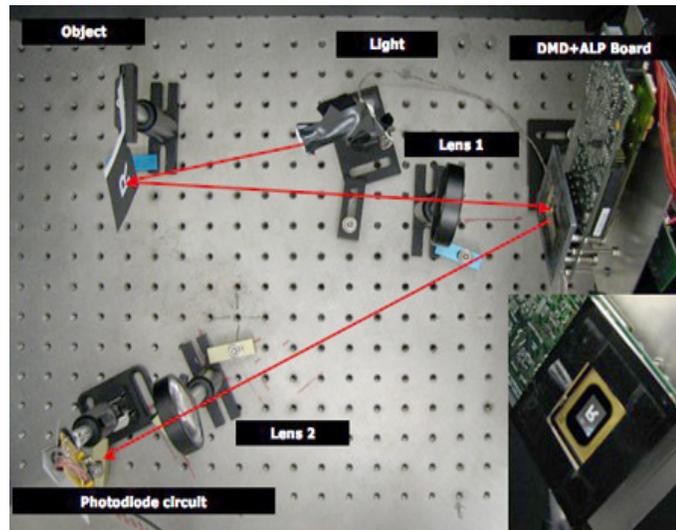


Figure 1.4: Rice University single pixel camera.

point and shoot cameras. Clearly, by using compressed sensing we can reduce the number of sensors and have smaller and cheaper cameras. The Rice University single pixel camera [48] and the MIT random lens imaging device [54] shown in Figure 1.4 and Figure 1.5 are two examples of such successful efforts.

1.3.3 Seismic data collection

Seismic data collection is an expensive and time consuming process. The ultimate goal here is to estimate the layers of the earth by measuring the reflections of a signal from different layers of the earth. Figure 1.6 depicts this process. The process starts with an explosion at the surface of the earth. The waves resulting from this explosion will be reflected from the boundaries of layers. There are many geo-phones to measure these reflections. This process is repeated many times and every time an explosion is made at a different location. By collecting all these reflections the final goal is to reconstruct the layers of the earth. This process is very time consuming because of the number of explosions. It turns out that compressed sensing has helped researchers reduce the number of explosions and make the whole process faster and cheaper. For more information refer to [100] and the references therein.



Figure 1.5: Random Lens imaging device.

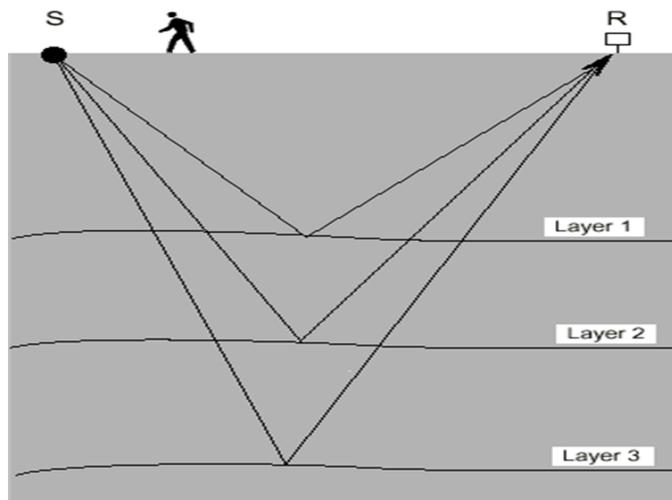


Figure 1.6: Seismic data collection procedure.

1.3.4 Other applications

The applications of compressed sensing are not just limited to the examples mentioned above. Compressed sensing has found applications in many other fields including but not limited to radar, analog to digital convertor design, electron tomography, machine learning, computational biology, astronomy, and We refer the interested reader to Rice University website at <http://dsp.rice.edu/cs>.

1.4 Organization and contribution of the thesis

1.4.1 Chapter 2

In Chapter 2 we review some of the existing sparse recovery algorithms and their theoretical guarantees. The theoretical results mentioned in this chapter are deterministic and can be applied as long as the matrices satisfy some conditions. These results are true whether or not the matrix is random. At the end of this chapter we use these results to draw some conclusions on the performance of these algorithms over some random matrix ensembles.

The main contribution of this chapter is the coherence proofs given for iterative hard thresholding and iterative soft thresholding algorithms. These results are based on [75]. Furthermore, we unify the existing deterministic results and provide the best bounds.

1.4.2 Chapter 3

Although the theoretical results of Chapter 2 are useful and provide guarantees on the performance of several algorithms, it has been shown in [10] that such performance bounds are pessimistic and hence weak for random matrices. Nevertheless, for comparison of different algorithms more accurate performance evaluations are needed. In chapter 3 we first introduce a weaker notion of correct recovery and based on that we introduce the maximin framework for tuning the parameters and comparing different compressed sensing algorithms. We then perform an extensive simulation to compare

the performance of some of the well-known algorithms in the compressed sensing literature. The results derived in this chapter are referred in the next chapters when we propose new sparse recovery algorithms. This chapter is based on the work that has been published in *IEEE journal of selected areas in signal processing* [76].

1.4.3 Chapter 4

In this chapter we will introduce a new class of sparse recovery algorithm called approximate message passing or AMP. We also introduce the state evolution framework as a flexible and very accurate analysis tool for AMP. This theoretical framework proves the exponential convergence rate (linear convergence in convex optimization literature) for the AMP algorithm which is the best known result for the compressed sensing algorithms. For more information, refer to Chapter 7. Furthermore, the connection between this algorithm and the ℓ_1 -minimization will be discussed. Using the maximin framework introduced in Chapter 3, we compare this algorithm with some of the other well-known algorithms and conclude that AMP “outperforms” the other algorithms.

This chapter is mainly based on the paper that has been published in *Proceedings of national academy of sciences* [38]. Some of the results mentioned in this chapter were published in [77].

1.4.4 Chapter 5

The goal of this chapter is to explain a general framework for deriving message passing algorithms. This chapter specially shows how one can design new approximate message passing algorithms, if more sophisticated priors are available on the signal. We will also generalize the state evolution framework. This chapter is based on the following two papers [40, 39].

1.4.5 Chapter 6

Measurement noise is an inevitable part of every compressed sensing system. In this chapter we consider the measurement noise and analyze the performance of our algorithm in the presence of noise. The connection between AMP and ℓ_1 -regularized least squares enables very accurate prediction of the noise sensitivity of the ℓ_1 -minimization. The results proved here will be compared with the corresponding results in the literature and as will be noticed these results are much more accurate and powerful. This chapter is based on [41].

1.4.6 Chapter 7

Recently several first-order algorithms have been suggested in the convex optimization and signal processing literatures for solving the ℓ_1 -penalized least square problem. In this chapter, we discuss these algorithms and their theoretical performance guarantees. As we will see, these algorithms perform much better on compressed sensing problems compared to their deterministic theoretical convergence rate. Therefore, for a more realistic study of these algorithms, we will explain the statistical framework and we explore the properties of the statistical convergence rate of different algorithms. Finally, using these properties we design several problem instances to compare the performance of these algorithms.

1.4.7 Chapter 8

In this chapter some of the open problems will be discussed. For each direction we will also review the results that exist in the literature to pave the way for those who want to solve these problems.

Chapter 2

Deterministic Results

In this chapter some of the algorithms for sparse recovery and compressed sensing are discussed. Two deterministic frameworks for analyzing these algorithms are explained and the guarantees these frameworks provide for the performance of each algorithm are mentioned.

As before, we assume that the signal $x_o \in \mathbb{R}^N$ is exactly sparse and $\|x\|_0 \leq k$. We observe n linear measurements of this vector through the matrix A , $y = Ax_o$. We assume that the columns of A have unit ℓ_2 norm.¹ For a subset of columns of A called J , A_J includes all the columns of A whose indices are in J , and x_J all the elements of x whose indices are in the set J . Finally, Υ_k^N represents the set of all N -dimensional vectors that are k -sparse.

Historically, the first tool that has been proposed for analyzing sparse recovery algorithms is the coherence [34]. The coherence of a matrix A is defined as

$$\mu = \max_{1 \leq i < j \leq N} |\langle A_i, A_j \rangle|. \quad (2.1)$$

For example, it is not difficult to see that the union of two orthonormal bases has coherence larger than $\frac{1}{\sqrt{N}}$. Also, for a general dictionaries of size D the coherence is lower bounded by $\mu \geq \sqrt{\frac{N-d}{d(N-1)}}$ [99]. The dictionaries that achieve this lower bound are equiangular dictionaries and every two elements have the same coherence.

¹We may relax this condition a little bit when we talk about random ensembles.

The second popular tool in compressed sensing is the Restricted Isometry Property or RIP² that was introduced in [20] and [19]. The matrix A satisfies $\text{RIP}(k, \gamma)$ if and only if

$$(1 - \gamma)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \gamma)\|x\|_2^2, \quad \forall x \in \Upsilon_k^N, \quad (2.2)$$

where $\gamma < 1$ is a fixed number. This condition implies that any $k \times k$ submatrix of the original matrix A is ‘near isometry’ and does not change the ℓ_2 -norm of a vector dramatically. One of the main disadvantages of the RIP condition is that for a given matrix checking the validity of RIP condition is an NP-complete problem itself. However, as will be discussed later in this chapter for random matrices that are of particular interest in compressed sensing, RIP provides useful performance guarantees.

In the rest of the chapter based on these two frameworks, we will summarize the best results for the correctness of some of the sparse recovery algorithms in the literature. Unfortunately, these sufficient conditions are not necessary and usually the constants are loose. In the next chapter we will define a notion of phase transition and discuss necessary and sufficient correct recovery conditions for different algorithms.

2.1 ℓ_1 -minimization

In the last chapter we mentioned that the most natural way for recovering exactly sparse signals is to solve ℓ_0 -minimization. Unfortunately, this problem is NP-complete and in general can not be solved by a polynomial time algorithm. Chen et al. [24] proposed the following convex optimization for recovering the sparsest solution;

$$(\mathcal{Q}_1) \quad \min \|x\|_1, \text{ s.t. } Ax = y. \quad (2.3)$$

This algorithm is called is called basis pursuit and is the convex relaxation of the ℓ_0 -minimization. This problem can be cast as a linear programming (LP) problem which in turn can be solved by interior point methods. The following theorem which

²It may also be called UUP for uniform uncertainty principle.

is due to [34] was the first formal result (for general measurement matrices) on the equivalence of basis pursuit and ℓ_0 -minimization.

Theorem 2.1.1. *For $k < \frac{1}{2}(\mu^{-1} + 1)$ the sparsest solution is unique and basis pursuit recovers that solution.*

As this theorem suggests, if the signal is sparse enough, then basis pursuit solves the NP-complete ℓ_0 -minimization. The next result which is due to Candés and Tao [20] proves the equivalence of ℓ_0 and ℓ_1 if the matrix A satisfies the RIP condition.

Theorem 2.1.2. *If matrix A satisfies $\text{RIP}(2k, \gamma)$ for $\gamma \leq \frac{3}{4+\sqrt{6}}$, then BP recovers the sparsest solution correctly when the sparsity is less than k .*

The proof of this theorem in this form is due to [59]. However it was proved before for $\gamma \leq \sqrt{2} - 1$ in [19] and with a different RIP condition in [20].

ℓ_1 -minimization is a convex optimization problem and can be solved in polynomial time. Several different schemes have been proposed for solving the ℓ_1 -minimization. These schemes range from interior point methods to homotopy schemes like LARS [49]. These methods are still computationally expensive and can not be used for very high dimensional problems. Therefore, researchers have suggested greedy approaches for recovering the sparsest solution. We will discuss some of these approaches that have deterministic performance guarantees in the next sections.³

2.2 Orthogonal matching pursuit (OMP)

Orthogonal matching pursuit is a greedy approach for finding the sparsest solution [23, 87]. It iteratively improves its estimate of the signal by choosing the column of a matrix that has the most correlation with the residual. More formally, OMP begins by setting the initial residual to y , the initial estimate of the signal to 0 and the active

³Recently first-order methods for solving ℓ_1 -minimization have attracted much attention because of their simplicity. They are related to iterative thresholding algorithms that will be discussed in this chapter but we postpone the formal discussion and analysis of these methods to Chapter 7.

set to \emptyset . Also set

$$r^0 = y \quad , \quad x^0 = 0 \quad \text{and} \quad I^0 = \emptyset. \quad (2.4)$$

At each step t , it first selects a new column of A according to

$$i_t^* \in \arg \max_i |\langle r^{t-1}, A_i \rangle|, \quad (2.5)$$

and adds it to the active set,

$$I^t = I^{t-1} \cup \{i_t^*\}. \quad (2.6)$$

Finally, it projects y on the range of A_{I^t} and updates the residual,

$$\begin{aligned} x^t &= (A_{I^t}^* A_{I^t})^{-1} A_{I^t}^* y, \\ r^t &= y - Ax^t. \end{aligned} \quad (2.7)$$

OMP was independently developed in [23] and [87]. The following theorem due to [104] provides a sufficient condition for the convergence of OMP in terms of coherence.

Theorem 2.2.1. *For $k < \frac{1}{2}(\mu^{-1} + 1)$ the sparsest solution is unique and OMP recovers that solution.*

It is also possible to provide sufficient RIP conditions for OMP [29].

Theorem 2.2.2. *If the matrix A satisfies $\text{RIP}(k + 1, \frac{1}{3\sqrt{k}})$ then the OMP algorithm is able to recover any k -sparse signal precisely after k iterations.*

The RIP condition provided for OMP is weaker than the RIP condition provided for ℓ_1 minimization. Several other greedy algorithms with similar ideas have been also proposed in the literature [33], [80], [27]. The main difference between these algorithms and the OMP is that instead of moving just one column to the active set at every iteration they add more columns to the active set. In addition to that some of these approaches allow removal of elements from the active set as well. We will discuss two of these algorithms [80] and [27] in a more general framework in the next chapter and compare their performance with other algorithms.

2.3 Iterative hard thresholding (IHT)

As we will see in the next chapter although greedy approaches such as OMP are faster than interior point or homotopy methods used in solving ℓ_1 -minimization, they are still slow since there is an inversion step involved in each step. A faster alternative approach is the family of iterative thresholding algorithms. Consider the hard threshold functions $\eta^H(x; \mu)$ to be applied elementwise to vectors: $\eta^H(x; \mu) = x\mathbb{I}_{\{|x|>\mu\}}$, where \mathbb{I} is the indicator function. Iterative hard thresholding (IHT) algorithm is defined with the following iteration:

$$x^{t+1} = \eta^H(x^t + A^*(y - Ax^t); \lambda^t); \quad (2.8)$$

where λ^t is the threshold value and x^t is the estimate of the sparse signal at time t . Note that the threshold value may depend on iteration t as well. The basic intuition is that since the solution satisfies the equation $y = Ax$, algorithm makes progress by moving in the direction of the gradient of $\|y - Ax\|^2$ and then promotes sparsity by applying a nonlinear thresholding function. This is depicted in Figure 2.1. One of the main challenges that affects the performance of iterative thresholding approaches is the way the threshold parameter is set. Here we consider one approach that fits well in the deterministic setting. In the next chapter, we will introduce other approaches that can be used for compressed sensing problems and are derived from statistical point of view. Suppose that an oracle tells us the true underlying k . Since the final solution is k sparse, the threshold can be set to the magnitude of the $(k + 1)^{\text{th}}$ largest coefficient. Iterative hard thresholding algorithm in this form was first introduced in [13]. However, this type of thresholding policy has also been used in [80],[27].

The iterations of IHT are the simplest among all the algorithms we have mentioned so far and the only operations that are needed for each iteration are multiplication of a vector by a matrix A or A^* . These operations can be performed efficiently for various measurement matrices such as sparse, partial Fourier, or partial DCT measurement matrices. The following two theorems prove the convergence of this algorithm to the correct result under certain conditions.

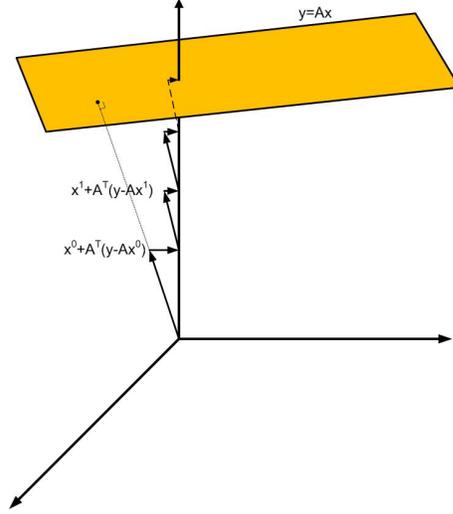


Figure 2.1: Geometric intuition of iterative hard thresholding algorithm. At every iteration the algorithm moves toward the hyperplane $y = Ax$ and then promotes sparsity by the hard thresholding function.

Theorem 2.3.1. *Suppose that $k < \frac{1}{3.1}\mu^{-1}$ and $\frac{|x_o(i)|}{|x_o(i+1)|} < 3^{\ell_i-4}, \forall i, 1 \leq i < k$. Then IHT finds the correct active set in at most $\sum_{i=1}^k \ell_i + k$ steps. After this step all of these elements will remain in the active set and the error will go to zero exponentially fast.*

The proof of this theorem is summarized in Appendix D.

Theorem 2.3.2. *Suppose that the matrix A satisfies $RIP(3k, \gamma)$ for $\gamma < 1$ and x_o is k sparse. Then the estimate given by IHT at time t satisfies*

$$\|x^t - x_o\|_2 \leq \gamma^{-t} \|x_o\|_2.$$

A more general version of this result was first proved in [13] for $\gamma \leq \frac{1}{\sqrt{8}}$. Later the bound on γ was improved in [59]. Since the proof of this theorem is very simple, we summarize it here.

Proof. The error at iteration $t + 1$ can be bounded by

$$\begin{aligned} \|x^{t+1} - x_o\| &= \|\eta^H(x^t + A^*(y - Ax^t) - x_o)\| = \|x_{I_t}^t - A_{I_t}^*(A_{I_t}x_{o_{I_t}} - A_{I_t}x_{I_t}^t - x_{o_{I_t}})\| \leq \\ &\|(I - A_{I_t}^*A_{I_t})(x^t - x_{o_{I_t}})\| \leq \delta_{3k}\|x^t - x_{o_{I_t}}\|, \end{aligned} \quad (2.9)$$

where I_t is the set that includes the indices of x_o, x^t, x^{t+1} and therefore its size is at most $3k$ and this fact accounts for the last inequality. By applying the same inequality inductively we obtain the inequality stated in the theorem. \square

Note 1: One of the main advantages of the coherence proof as we will see in the next section is the easy extension of this approach to other iterative thresholding algorithm. Another advantage is that it provides more detailed information on the performance of the algorithm. For example, the number of iterations needed for recovering the active set correctly and so on.

Note 2: Although the results that are provided in this algorithm are slightly weaker than the results provided for ℓ_1 -minimization, in practice and specially in compressed sensing (as we will see in the next section), the algorithm performs much worse than ℓ_1 in the sparsity-measurement tradeoff. We will formally state this in the next chapter.

2.4 Iterative soft thresholding (IST)

Another iterative thresholding algorithm that will play an important role in our discussion in the next few chapters is iterative soft thresholding algorithm. Consider the soft thresholding function $\eta_\mu^S(x) = (|x| - \mu)_+$ where $(a)_+$ is equal to a if $a > 0$ and is equal to zero otherwise. Then iterative soft thresholding algorithm is given by the following iteration:

$$x^{t+1} = \eta^S(x^t + A^*(y - Ax^t); \lambda^t); \quad (2.10)$$

Clearly the iteration looks very similar to the iteration of iterative hard thresholding algorithm. The only difference is the usage of soft thresholding function instead of hard thresholding. The main advantage of using soft thresholding in the iteration is the connection of soft thresholding function with ℓ_1 -minimization. This connection

will be explored in detail in Chapter 7. However, to provide some intuition on the soft thresholding function it is useful to gloss over some of those results. Consider the following optimization problem:

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} \|z - x\|_2^2 + \lambda \|x\|_1. \quad (2.11)$$

It is straightforward to confirm that the function is minimized at $x = \eta^S(z; \lambda)$. The soft thresholding function is called the “proximity operator” of the ℓ_1 -norm and this suggests iterative soft thresholding algorithm with a fixed threshold parameter for solving the following problem,

$$(\mathcal{Q}_\lambda) \quad \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1. \quad (2.12)$$

Refer to Chapter 7 for more information. Another interesting fact about the iterative soft thresholding algorithm is that, if A is orthonormal and $A^*A = I$, then the solution of \mathcal{Q}_1 will be $\eta^S(A^*y; \lambda)$.

Theorem 2.4.1. *Suppose that $k < \frac{1}{4.1}\mu^{-1}$ and $\forall i, 1 \leq i < k$, we have $\frac{|x_o(i)|}{|x_o(i+1)|} < 2^{\ell_i - 5}$. Then IST recovers the correct active set in at most $\sum_{i=1}^k \ell_i + k$ steps. After that all these coefficients will remain in the active set and the error will go to zero exponentially fast.*

The proof of this theorem is summarized in Appendix D. The number of iterations needed to recover the active set, depends on the ratio of the coefficients in IST and IHT. But, this dependency is roughly logarithmic and therefore it works pretty well in practice even in case of high dynamic range signals. Also, the algorithms find the correct active set in a finite number of iterations and once they find the correct active set, they will converge to the exact solution exponentially fast.

2.5 Connection between RIP and coherence

So far we have considered the RIP conditions and coherence conditions as two separate conditions for sparse recovery problems. The following simple lemma explain this

connection.

Lemma 2.5.1. *If the coherence of matrix A is bounded above by μ and the columns of A are normalized, then for any $k < \mu^{-1} + 1$, A satisfies $RIP(k, (k-1)\mu)$.*

This lemma is a very simple application of Grešgorin's theorem [66]. However, as we will see later, the coherence condition provides weaker results compared to RIP when dealing with random matrices. The main advantage of coherence is that it is very easy to calculate for a given matrix and can be used for deterministic settings while calculating RIP constant is an NP-complete problem itself.

2.6 Implications for compressed sensing

In compressed sensing the measurement matrices are drawn from some random ensemble. In this section we aim to show the implications of coherence and RIP conditions for random measurement matrices.

2.6.1 Coherence of random measurement matrices

The goal of this section is to derive upper bounds for the coherence of different random matrix ensembles. Since our definition of the coherence is for matrices with unit length columns, we normalize each column with its ℓ_2 -norm.

Random sign ensemble matrix

Suppose that the elements of A are drawn iid from the following distribution:

$$A_{i,j} \sim \begin{cases} \frac{1}{\sqrt{n}}, & \text{w.p. } 0.5, \\ \frac{-1}{\sqrt{n}}, & \text{w.p. } 0.5. \end{cases} \quad (2.13)$$

This ensemble is one of the standard ensembles in CS and we call it Random Sign Ensemble or RSE. The following well-known concentration result on a sequence of bounded iid random variables is very useful in our discussion.

Lemma 2.6.1. (*Hoeffding Inequality*) Suppose that X_1, X_2, \dots, X_n are independent random variables each bounded between $[a_i, b_i]$.

$$\begin{aligned} P\{S_n - E(S_n) \geq t\} &\leq \exp\left(\frac{-2t^2}{\sum_i^n (b_i - a_i)^2}\right), \\ P\{E(S_n) - S_n \geq t\} &\leq \exp\left(\frac{-2t^2}{\sum_i^n (b_i - a_i)^2}\right). \end{aligned} \quad (2.14)$$

The interested reader may find the proof in most of the standard probability text books such as [91].

Theorem 2.6.2. [105] *If the elements of the matrix A are iid drawn from the RSE ensemble then the coherence of the matrix is less than $\sqrt{\frac{4}{n} \ln \frac{N}{\epsilon}}$ with probability larger than $1 - \epsilon^2$.*

Proof. According to the union bound we have,

$$\mathbb{P}(\max_{i < j} |\langle A_i, A_j \rangle| > x) \leq \frac{N(N-1)}{2} \mathbb{P}(|\langle A_i, A_j \rangle| > x).$$

Also Hoeffding inequality implies that

$$\mathbb{P}(|\langle A_i, A_j \rangle| > x) \leq 2e^{-\frac{nx^2}{2}}.$$

Plugging in $\sqrt{\frac{4}{n} \ln \frac{N}{\epsilon}}$ for x , gives the desired bound. \square

Gaussian measurement matrix

Suppose that the elements of Φ are drawn iid from $N(0, \frac{1}{n})$. We normalize each column by its ℓ_2 -norm. This ensemble is called uniform spherical ensemble since each column is drawn iid from Haar measure on unit sphere. We call the final measurement matrix A . The goal is to find an upper bound on the coherence of this matrix. Clearly, since the matrices are random we expect the bound to hold with high probability. Let us start with the following well-known Gaussian tail bound.

Lemma 2.6.3. *For $X \sim N(0, \sigma^2)$, $\mathbb{P}(|X| \geq x) \leq 2e^{-\frac{x^2}{2\sigma^2}}$.*

Proof.

$$\mathbb{P}(X \geq x) = \mathbb{P}(e^{\lambda X} \geq e^{\lambda x}) \leq e^{-\lambda x} \mathbb{E}[e^{\lambda X}] = e^{-\lambda x} e^{\frac{\lambda^2 \sigma^2}{2}}$$

$\lambda^* = \frac{x}{\sigma^2}$ minimizes the right hand side and the minimum value is $e^{\frac{-x^2}{2\sigma^2}}$ QED. \square

Theorem 2.6.4. *Let the columns of the matrix A be iid samples from Haar measure on unit sphere; If $n \geq 200 \ln(N\sqrt{3}/\epsilon)$, then the coherence of the matrix is less than $\sqrt{\frac{9}{n} \ln \frac{\sqrt{2N}}{\epsilon}}$ with probability larger than $1 - \epsilon^2$.*

Proof. In this proof we call the original Gaussian matrix Φ and the actual matrix which is the column normalized version of Φ , A .

Step 1: ℓ_2 -norm of each column of Φ is concentrated around 1. The first question to address is to calculate the norms of the columns of Φ . We prove

$$\begin{aligned} \mathbb{P}(\exists i : \|\Phi_i\|_2^2 - 1 \geq t) &\leq N e^{-n/2(t - \ln(1+t))}, \\ \mathbb{P}(\exists i : \|\Phi_i\|_2^2 - 1 \leq -t) &\leq N e^{n/2(t + \ln(1-t))}. \quad \forall t \in (0, 1). \end{aligned} \quad (2.15)$$

Here we just prove the first bound. The proof of the second one follows exactly the same lines and therefore is skipped here.

$$\begin{aligned} \mathbb{P}(\exists i : \|\Phi_i\|_2^2 - 1 \geq t) &\leq N \mathbb{P}(\|\Phi_1\|_2^2 - 1 \geq t) = N \mathbb{P}(e^{\lambda(\sum_{i=1}^n \Phi_{1i}^2 - 1)} \geq N e^{\lambda t}) \\ &\leq N e^{-\lambda t} \mathbb{E}(e^{\lambda(\sum_{i=1}^n \Phi_{1i}^2 - 1)}) = N e^{-\lambda t - \lambda} \left(\mathbb{E}(e^{\lambda \Phi_{1i}^2}) \right)^n \\ &= N e^{-\lambda t - \lambda} \left(\frac{1}{1 - \frac{2\lambda}{n}} \right)^{\frac{n}{2}}. \end{aligned}$$

$\lambda^* = \frac{nt}{2(t+1)}$ minimizes the last bound and proves (2.15). Substituting a value of t in (2.15) proves

$$\begin{aligned} \mathbb{P}(\|\Phi_i\|_2^2 < 0.81) &\leq e^{-0.01n}, \\ \mathbb{P}(\|\Phi_i\|_2^2 > 1.25) &\leq e^{-0.01n}. \end{aligned} \quad (2.16)$$

Step 2: Correlations between the columns. From the union bound we have

$$\mathbb{P}(\max_{i < j} |\langle A_i, A_j \rangle| > x) \leq \frac{N(N-1)}{2} \mathbb{P}(|\langle A_i, A_j \rangle| > x).$$

The right hand side of the above equation can also be bounded by using Lemma 2.6.3 and (2.16).

$$\begin{aligned} \mathbb{P}(|\langle A_i, A_j \rangle| > x) &= \mathbb{P}\left(\frac{|\langle \Phi_i, \Phi_j \rangle|}{\|\Phi_i\| \|\Phi_j\|} > x\right) = \\ &\mathbb{P}\left(\frac{|\langle \Phi_i, \Phi_j \rangle|}{\|\Phi_i\| \|\Phi_j\|} > x \mid \|\Phi_i\|_2 > .9, \|\Phi_j\|_2 > .9\right) \mathbb{P}(\|\Phi_i\|_2 > .9, \|\Phi_j\|_2 > .9) + \\ &\mathbb{P}\left(\frac{|\langle \Phi_i, \Phi_j \rangle|}{\|\Phi_i\| \|\Phi_j\|} > x \mid \|\Phi_i\|_2 < .9 \vee \|\Phi_j\|_2 < .9\right) \mathbb{P}(\|\Phi_i\|_2 < .9 \vee \|\Phi_j\|_2 < .9) \leq \\ &\mathbb{P}(|\langle \Phi_i, \Phi_j \rangle| > .81x \mid \|\Phi_i\|_2 > .9, \|\Phi_j\|_2 > .9) \mathbb{P}(\|\Phi_i\|_2 > .9, \|\Phi_j\|_2 > .9) + \\ &\mathbb{P}(\|\Phi_i\|_2 < .9) + \mathbb{P}(\|\Phi_j\|_2 < .9) \leq \mathbb{P}(|\langle \Phi_i, \Phi_j \rangle| > .81x) + 2\mathbb{P}(\|\Phi_i\|_2 < .9) = \\ &\mathbb{E}(\mathbb{P}(|\langle \Phi_i, \Phi_j \rangle| \geq .81x \mid \Phi_j)) + 2\mathbb{P}(\|\Phi_i\|_2 < .9) = \mathbb{E}(e^{\frac{-.3nx^2}{\|\Phi_j\|_2^2}}) + 2\mathbb{P}(\|\Phi_i\|_2 < .9) \leq \\ &2e^{-.24nx^2} \mathbb{P}(\|\Phi_j\|_2^2 \leq 1.25) + \mathbb{P}(\|\Phi_j\|_2^2 \geq 1.25) + 2\mathbb{P}(\|\Phi_i\|_2 < .9) \leq \\ &2e^{-.24nx^2} + 3e^{-0.01n}. \end{aligned}$$

After plugging x in, it is easy to see that this probability is less than ϵ^2 . \square

Note: From the proof of this theorem it is clear that we have not been very careful with the constants. By changing the conditions in the above derivation one may obtain better results. However in this chapter we just care to find the orders right and not the constants. The constants are of course very important for engineering applications as well and we will discuss this issue in detail in the next chapter.

Coherence of sub-gaussian matrices

The results mentioned for Gaussian and RSE matrices can also be extended to a more general class of matrices which are called sub-gaussian matrices.

Definition 2.6.1. *A random variable X is called sub-gaussian iff $\mathbb{P}(|X| > t) \leq C \exp(-ct^2)$ for two constants c and C . Also a random variable Y is called sub-exponential if and only if $\mathbb{P}(|Y| > t) \leq C \exp(-ct)$ for two constants c and C .*

Clearly, every bounded random variable is sub-gaussian. Gaussian random variable is also sub-gaussian and exponential and Laplace random variables are sub-exponential. Some of the standard properties of sub-gaussian and sub-exponential random variables are summarized in Appendix A.

Lemma 2.6.5. *Suppose that X and Y are two independent sub-gaussian random variables with constants C_x, c_x, C_y, c_y . Then X^2 and XY are sub-exponential random variables.*

Proof. The proof of X^2 is immediate, therefore we just prove the second one.

$$\begin{aligned} \mathbb{P}(|XY| > t) &= \int \mathbb{P}\left(|X| > \frac{t}{|y|}\right) dF_Y(y) \leq \int C_x e^{-c_x \frac{t^2}{|y|^2}} dF_Y(y) \\ &= \int_{|y| \geq \sqrt{t}} C_x e^{-c_x \frac{t^2}{|y|^2}} dF_Y(y) + \int_{|y| \leq \sqrt{t}} C_x e^{-c_x \frac{t^2}{|y|^2}} dF_Y(y) \\ &\leq \int_{|y| \geq \sqrt{t}} C_x dF_Y(y) + \int_{|y| \leq \sqrt{t}} C_x e^{-c_x t} dF_Y(y) \leq C_x C_y e^{-c_y t} + C_x e^{-c_x t}. \end{aligned}$$

It is also possible to find a symmetric upper bound for this probability by switching the role of X and Y and take the average of the two upper bounds. \square

Theorem 2.6.6. *Suppose that the elements of the matrix are drawn iid from a sub-gaussian distribution with mean zero and variance $1/n$. There exists a constant c for which the coherence of this matrix satisfies the following property.*

$$\mathbb{P}\left(\mu < \sqrt{\frac{1}{nc} \log \frac{N^2}{\epsilon}}\right) \geq 1 - \epsilon. \quad (2.17)$$

Proof. Since most parts of the proof are very similar to the corresponding parts in the proof of the Gaussian matrix, I just mention the differences. Suppose that Φ represents the matrix before the column normalization. Then, we have

$$\mathbb{P}\left(\sum_j \Phi_{1j} \Phi_{2j} > t\right) \stackrel{(1)}{=} \mathbb{P}\left(\frac{1}{n} \sum_j Z_j > t\right) \stackrel{(2)}{\leq} e^{-cnt^2}.$$

For equality (1) we assume that $Z_j = \Phi_{1j} \Phi_{2j}$ and therefore, Z_j is a sub-exponential random variable. Inequality (2) is the result of Theorem A.0.5 and we assume that

$t < 1$. Because of symmetry $\mathbb{P}(\sum_j \Phi_{1j} \Phi_{2j} < -t) \leq e^{-cnt^2}$ as well. Another important property that was necessary for the proof of the Gaussian matrix was the concentration of the column norm. We skip the proof of this as well since it is very similar to what was explained here. \square

How many measurements?

So far we have considered the coherence of several ensembles. Apart from the constants for all these ensembles $\mu \sim \sqrt{\frac{\ln N}{n}}$. It is also important to notice that all the algorithms mentioned in this chapter are able to recover the sparsest solution, if the sparsity level is at the order of μ^{-1} . Therefore the final conclusion is that in case of random matrices, coherence condition proves that the exact recovery happens with high probability if $k \sim \sqrt{\frac{n}{\ln N}}$ or in other words the number of measurements that we need is $n \sim k^2 \ln N$. As we will see in the next section RIP provides stronger results for random matrices.

2.6.2 Analysis of RIP for random matrices

In this section we consider $n \times N$ random matrices. We assume that elements are normalized in a way that the norm of each column is concentrated around 1 and then we find an upper bound for the RIP constants.

Gaussian ensemble

The first step in finding the RIP constants for random matrices is the concentration of measure for functions of random vectors. Here is a well-known concentration of measure theorem for Gaussian random vectors.

Theorem 2.6.7. [101] *Let $x \in \mathbb{R}^n$ be a vector of $N(0, 1)$ random variables and f be a Lipschitz function, i.e.,*

$$|f(x) - f(y)| \leq L\|x - y\|_2. \quad (2.18)$$

Then, for any $t > 0$ we have

$$\mathbb{P}(f(x) - \mathbb{E}f(x) > t) \leq e^{\frac{-t^2}{2L^2}}.$$

Theorem 2.6.8. *For two matrices A and B we have, $|\sigma_{\max}(A) - \sigma_{\max}(B)| \leq \|A - B\|_F$ and $|\sigma_{\min}(A) - \sigma_{\min}(B)| \leq \|A - B\|_F$. Here σ_{\max} represents the largest eigenvalue of a matrix and σ_{\min} the smallest one.*

Proof.

$$|\sigma_{\max}(A) - \sigma_{\max}(B)| \leq |\sigma_{\max}(A - B)| \leq \|A - B\|_F.$$

For the smallest eigenvalue we also have

$$\begin{aligned} \sigma_{\min}(A) &= \inf_{u \in S^{n-1}} \|Au\|_2 = \inf_{u \in S^{n-1}} \|(A - B)u + Bu\|_2 \leq \inf_{u \in S^{n-1}} \|(A - B)u\|_2 + \|Bu\|_2 \\ &\stackrel{(1)}{\leq} \inf_{u \in S^{n-1}} \|A - B\|_F + \|Bu\|_2 = \|A - B\|_F + \sigma_{\min}(B). \end{aligned}$$

In the above derivation $S^{n-1} = \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$ and inequality (1) is due to the Cauchy-Schwartz. \square

A more general version of this result is due to Weyl [66]. This theorem states that both the smallest singular value and the largest singular value are Lipschitz functions of the matrix and according to Theorem 2.6.7 they concentrate around their means. Therefore, the main challenge is to calculate their means. We use the following theorem to calculate a bound for the expected value of singular values which is known as Gordon Inequality [71]. A special form of the Gordon Inequality is also called Slepian inequality.

Theorem 2.6.9. *Let $X_{u,v}$ and $Y_{u,v}$ be centered Gaussian processes. Also assume that,*

1. $\|X_{u,v} - X_{u',v'}\|_2 \leq \|Y_{u,v} - Y_{u',v'}\|_2$ if $u \neq u'$;
2. $\|X_{u,v} - X_{u,v'}\|_2 = \|Y_{u,v} - Y_{u,v'}\|_2$.

Then $\mathbb{E} \sup_{u \in U} \inf_{v \in V} X_{u,v} \leq \mathbb{E} \sup_{u \in U} \inf_{v \in V} Y_{u,v}$.

For the proof of this theorem refer to [71].

Theorem 2.6.10. *Suppose that the elements of matrix A are iid $N(0, \frac{1}{n})$. The minimum and maximum singular values of this matrix satisfy the following inequality:*

$$1 - \sqrt{\frac{N}{n}} \leq \mathbb{E}\sigma_{\min}(A) \leq \mathbb{E}\sigma_{\max}(A) \leq 1 + \sqrt{\frac{N}{n}}.$$

Proof. We start with the definition of the largest and smallest singular values,

$$\begin{aligned} \sigma_{\max}(A) &= \sup_{u \in S^{N-1}, v \in S^{n-1}} \langle Au, v \rangle, \\ \sigma_{\min}(A) &= \inf_{u \in S^{N-1}} \sup_{v \in S^{n-1}} \langle Au, v \rangle. \end{aligned}$$

The goal is to use the Gordon inequality. Therefore we should construct another Gaussian process with the desired properties. Suppose that $g \sim N(0, \frac{1}{n}I_n)$ and $h \in N(0, \frac{1}{N}I_N)$ are two Gaussian vectors. Define $Y_{u,v} = \langle h, u \rangle + \langle g, v \rangle$ and $X_{u,v} = \langle Au, v \rangle$. Using the fact that $\|u\|_2 = 1$ and $\|v\|_2 = 1$, it is straightforward to prove that all the conditions of the Gordon inequality hold for $X_{u,v}$ and $Y_{u,v}$. Using the Gordon inequality the proof of the upper bound is immediate.

$$\begin{aligned} \mathbb{E} \sup_{u \in S^{N-1}, v \in S^{n-1}} \langle Au, v \rangle &= \mathbb{E} \sup_{u \in S^{N-1}, v \in S^{n-1}} Y_{u,v} \leq \mathbb{E} \sup_{u \in S^{N-1}} \langle h, u \rangle + \mathbb{E} \sup_{v \in S^{n-1}} \langle g, v \rangle \stackrel{(1)}{\leq} \\ \mathbb{E} \|g\|_2 + \mathbb{E} \|h\|_2 &\stackrel{(2)}{\leq} 1 + \sqrt{N/n}, \end{aligned}$$

where (1) is the result of Cauchy-Schwartz inequality and (2) is derived from Jensen's inequality. This form of the Gordon inequality is known as Slepian inequality.

For the lower bound on the smallest singular value we should just use $-X_{u,v}$ and $-Y_{u,v}$ in the Gordon inequality and use similar arguments. \square

The following result can be easily proved by combining the results of Theorems 2.6.10 and 2.6.7.

Theorem 2.6.11. *Suppose that the elements of an $n \times N$ matrix A are drawn from*

$N(0, 1/n)$. Then

$$\begin{aligned} \mathbb{P} \left(\sup_{T:|T|<\alpha k} \sigma_{\max}(A_T) > 1 + \sqrt{\frac{\alpha S}{n}} + t \right) &\leq \binom{N}{\alpha k} e^{-n \frac{t^2}{2}}, \\ \mathbb{P} \left(\inf_{T:|T|<\alpha k} \sigma_{\min}(A_T) < 1 - \sqrt{\frac{\alpha S}{n}} - t \right) &\leq \binom{N}{\alpha k} e^{-n \frac{t^2}{2}}. \end{aligned}$$

To see the implication of this theorem for the compressed sensing problem it is just enough to use the Stirling approximation for $\binom{N}{\alpha k}$ term $e^{\alpha k \ln N/\alpha k} e^{-n \frac{t^2}{2}}$. Therefore, if we set t such that $1 + \sqrt{\frac{\alpha k}{n}} + t \leq \sqrt{1 + \beta}$ and $1 + \sqrt{\frac{\alpha k}{n}} - t \leq \sqrt{1 - \beta}$, and if $n \gtrsim \alpha k \ln \frac{N}{\alpha k}$ the probability that this class of matrices satisfies $RIP(\alpha k, \beta)$ goes to 1 exponentially fast and the number of measurements needed is logarithmically proportional to the dimension of the signal.

Sub-gaussian ensemble

In the last section we discussed the results for Gaussian random matrices. Similar results can be proved for sub-gaussian random matrices. However the results are a little bit weaker for this general class as will be shown in this section. We start with Dudley's integral inequalities that will be used throughout this section. For the proof of this theorem refer to [108].

Theorem 2.6.12. *Consider a random process X_t where t belongs to a compact set T of a given metric space (\mathcal{M}, d) . Suppose that this process satisfies the following properties,*

1. $E(X_t) = 0 \quad \forall t$.
2. $\mathbb{P}(|X_t - X_s| > ud(t, s)) \leq Ce^{-cu^2} \quad \forall u > 0$.

Then,

$$\mathbb{E} \sup_{t \in T} X_t \leq C \int_0^\infty \sqrt{\log N_\epsilon(T)} d\epsilon,$$

where $N_\epsilon(T)$ is the minimum number of elements in an ϵ covering of the space T .

If we replace the sub-gaussian tail in the second condition with sub-exponential tail we get another version of Dudley's inequality which is,

$$\mathbb{E} \sup_{t \in T} X_t \leq C \int_0^\infty \log N_\epsilon(T) d\epsilon.$$

We can now prove the following theorem for general sub-gaussian matrices.

Theorem 2.6.13. *There exist two constants c, C such that,*

$$\begin{aligned} \mathbb{P}\left(\sup_{T:|T|<\alpha k} \sigma_{\max}(A_T) > 1 + c\sqrt{\frac{\alpha S}{n}}\right) &\leq \binom{N}{\alpha k} e^{-Cn} \\ \mathbb{P}\left(\inf_{T:|T|<\alpha k} \sigma_{\min}(A_T) > 1 - c\sqrt{\frac{\alpha S}{n}}\right) &\leq \binom{N}{\alpha k} e^{-Cn} \end{aligned}$$

Proof. We do not intend to give the complete proof here and we refer the interested reader to [108]. We just highlight the main tricks in the proof that may help readers to prove it for themselves. The first step is of course the union bound as before. Therefore we just need an upper bound for one submatrix of size $n \times \alpha k$. Consider the following random process on unit sphere $S^{\alpha k - 1}$.

$$X_x = \left| \frac{1}{n} \|Ax\|_2^2 - 1 \right|.$$

The next step is to prove that this process is sub-exponential and use the second form of Dudley's inequality to prove the theorem. \square

Note: As mentioned before the result is weaker than the Gaussian results since $\sigma_{\max}(A) < 1 + c\sqrt{\frac{\alpha k}{n}}$ with high probability and the constant c is always bigger than 1. However for Gaussian ensembles we could prove similar result with $c = 1$.

How many measurements?

Form the above arguments it is clear that if we ignore the constants, the RIP condition implies that if the number of measurements is $m \sim k \log N/k$ then the algorithms we mentioned before recover the correct answer with overwhelmingly high probability. If

we compare this with what we achieved through coherence conditions we see that the bounds provided with RIP are stronger. The order of the bounds derived from RIP for random matrices are optimal. However, in engineering applications the constants are as important. Therefore, there is a need for deriving exact sampling theorems with all the constants specified carefully. In the next chapter we explain the concept of phase transition that gives us an accurate and exact sampling theorem for an algorithm. Unfortunately, because of the more difficult nature of the phase transition phenomena much less effort has been devoted to this fundamental problem of compressed sensing compared to the RIP or coherence conditions. This thesis provides a new approach for measuring phase transition for a class of algorithms. We hope that extensions of this approach may shed more light on the phase transitions of different algorithms in the near future.

2.7 Other algorithms

Developing new algorithms for sparse recovery is a very active area of research and new algorithms are proposed monthly. Unfortunately, it is almost impossible to go over all the algorithms and compare their theoretical guarantees and their actual performance. We will discuss some of these algorithms in the next chapters. Specially we will devote one chapter to the first-order methods that have been proposed for solving ℓ_1 -minimization and comparison of those algorithms with the algorithms proposed in this thesis. However, the first step is to propose a unified framework for comparing different compressed sensing algorithms which is the scope of the next chapter.

Chapter 3

Maximin Framework

As mentioned in the previous chapter numerous schemes have been proposed for obtaining sparse solutions of underdetermined systems of linear equations; Popular methods have been developed from many viewpoints: ℓ_1 -minimization [24, 35, 104, 49, 70, 47], matching pursuit [78, 87, 105, 33], iterative thresholding methods [92, 28, 96, 97, 52, 65, 50, 11, 58, 80, 12, 13], subspace methods [33, 81, 80, 27], convex regularization [26, 115] and nonconvex optimization [63, 53, 22]. The specific proposals are often tailored to different viewpoints, ranging from formal analysis of algorithmic properties [28, 105, 62, 33, 47], to particular application requirements [96, 97, 52]. Such algorithms have potential applications in fields ranging from medical imaging to astronomy [73, 15].

The potential user now has a bewildering variety of ideas and suggestions that might be helpful, but this, paradoxically creates uncertainty and may cause said potential user of such algorithms to avoid the topic entirely.

The goal of this chapter is to develop a common framework for comparing properties of reconstruction algorithms in compressed sensing. It considers several popular iterative thresholding algorithms, and abstracts them into a few different types, each with tunable parameters. It defines a quantitative notion of performance and after a large-scale computational study, identifies a tuned version of each algorithm type offering the best performance guarantee across a universe of test suites. Performance is measured by the undersampling-sparsity tradeoff, or ‘phase-transition curve’, and

for each algorithm type, we identify the optimally tuned instance that maximizes the worst-case tradeoff, yielding one recommended algorithm for each algorithm type.

The empirical tuning approach has a larger significance for the field of sparse representations and compressed sensing. Many of the better known papers in this field discuss *what can be proved rigorously*, using mathematical analysis. Often, what can be proved is weaker than happens in practice. For practical engineering applications it is more important to know *what really happens* rather than what can be proved. Empirical studies provide a direct method to give engineers useful guidelines about what really does happen.

To make this chapter self sufficient we first review the notations. An unknown vector $x_o \in \mathbb{R}^N$ is of interest; we have measurements $y = Ax_o$. Here A is an $n \times N$ matrix and $N > n$. Although the system is underdetermined, it has been shown that, when it exists, sufficient *sparsity* of x_o may allow unique identification of x_o . We say that x_o is k -sparse if it has at most k nonzeros.

3.1 Phase transitions

In the case of ℓ_1 -minimization with A a random matrix, there is a well-defined ‘break-down point’: ℓ_1 can successfully recover the sparsest solution provided k is smaller than a certain definite fraction of n .

Let $\delta = n/N$ be a normalized measure of problem indeterminacy and let $\rho = k/n$ be a normalized measure of the sparsity. We obtain a two-dimensional phase space $(\delta, \rho) \in [0, 1]^2$ describing the difficulty of a problem instance – problems are intrinsically harder as one moves up and to the left. Displays indicating success and failure of ℓ_1 -minimization as a function of position in phase space often have an interesting two-phase structure (as shown in Figure 3.1), with phases separated by the curve $(\delta, \rho_{\ell_1}(\delta))$, for a specific function ρ_{ℓ_1} .

Let A be a random matrix with iid Gaussian entries and let $y = Ax_0$ with x_0 k -sparse. In [32, 45] one can find explicit formulas for a function ρ definable with the aid of polytope theory and having the following property. *Fix $\epsilon > 0$. The probability that (P_1) recovers the sparsest solution to $y = Ax$ tends to 0 or 1 with increasing*

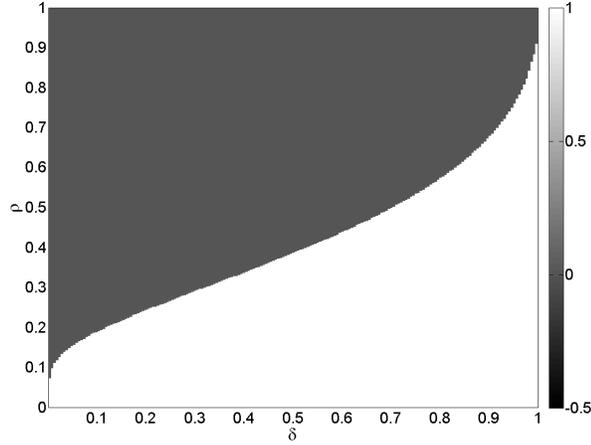


Figure 3.1: *Phase Diagram for ℓ_1 minimization.* horizontal axis: indeterminacy $\delta = n/N$. vertical axis: sparsity $\rho = k/n$. Shaded attribute depicts limiting probability that ℓ_1 successfully reconstructs the sparsest solution, as a function of the state variables (ρ, δ) . Dark region: limiting probability 0. Light region: limiting probability is 1.

system size according as $k \sim n \cdot (\rho_{\ell_1}(n/N) \pm \epsilon)$. Informally, all that matters is whether $(n/N, k/n)$ lies above or below the curve $(\delta, \rho_{\ell_1}(\delta))$. This is the conclusion of a rigorously proven theorem that describes asymptotic properties as $N \rightarrow \infty$; it also describes what actually happens at finite problem sizes [43]. The empirically observed fraction of successful recoveries decays from one to zero as the problem sparsity $\rho = k/n$ varies from just below the critical level $\rho_{\ell_1}(\delta)$ specified in theory to just above it. This transition zone is observed to get increasingly narrow as N increases, matching the theorem, which says that in the large N limit, the zone has vanishing width. Similar phenomena have been either observed or proved for other algorithms as well. To explain it more formally here, suppose that the elements of x_o are drawn from a given distribution $F_X(x_{oi}) = (1 - \epsilon)\delta_0(x_{oi}) + \epsilon G(x_{oi})$, where δ_0 is equal to 1 when x_{oi} is zero and 0 otherwise. $G(x_{oi})$ is another distribution function that specifies the distribution of the non-zero elements in the vector but it is usually unknown beforehand. In other words each element of x_o is zero with probability $1 - \epsilon$ and is drawn from another distribution G with probability ϵ . If the dimension of x_o is

N then the expected number of non-zero elements is $N\epsilon$ and according to Hoeffding inequality the number of non-zeros will concentrate around this value pretty rapidly as $N \rightarrow \infty$. According to the compressed sensing framework x_o is measured through a fat random matrix A whose elements are drawn iid from some distribution. Suppose that algorithm \mathcal{A} is used for recovering the original vector from the measurements y . No matter how well \mathcal{A} works there are certain samples of this random ensemble on which \mathcal{A} has no chance of recovering the correct answer. For example when A matrix is equal to zero or when x_o is dense. Therefore an important measure of performance is the probability of correct recovery of the algorithm as a function of (N, δ, ρ, F) . If the distribution F is fixed, the probability of correct recovery is considered as a function of (N, ρ, δ) . Since in compressed sensing we are interested in very high dimensional vectors, we calculate these probabilities as $N \rightarrow \infty$. This results in a 2-D map of the probability of correct recovery in terms of (ρ, δ) . The mapping is called the phase diagram. Figure 3.1 depicts the phase diagram for the ℓ_1 minimization algorithm. As explained before the probability of correct recovery exhibits a sharp transition i.e. there is a curve $\rho^*(\delta)$ below which the ℓ_1 minimization recovers the correct answer with probability 1 and above which the probability of correct recovery goes to zero very rapidly as $N \rightarrow \infty$. The existence of phase transition for many other algorithms has been either empirically observed or theoretically proved. As an example Figure 3.2 summarizes the fact for the IHT algorithm the transition becomes sharper and sharper as we increase the dimension of the problem. The interested reader may refer to these papers for more information [33, 105, 47, 44, 32, 45, 46]. Phase transition displays the sparsity-measurement trade-off for an algorithm. Therefore it is useful for comparison and tuning purposes. The only problem is the dependence of the phase transition on the distribution of the input coefficients which is usually not known in practice. This issue is addressed in the next section.

3.1.1 Maximin tuning

Consider a sparse recovery algorithm \mathcal{A}_θ . θ includes all the free parameters of the algorithm. It is clear that the performance of this algorithm depends highly on the

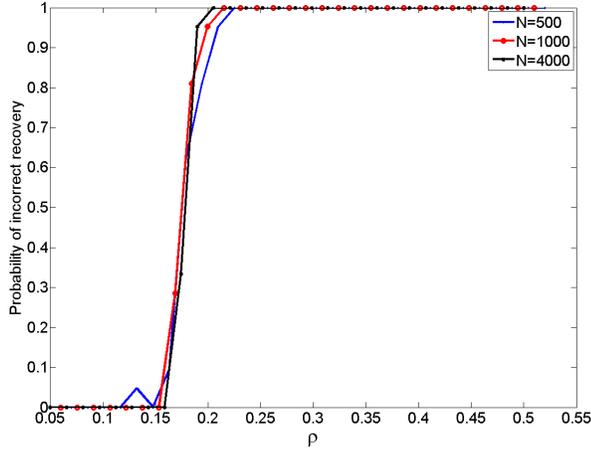


Figure 3.2: *Finite- N Phase Transitions*. Fraction of unsuccessful recovery attempts by IHT, as a function of sparsity level ρ . Here $\delta = .5$ and $\rho = k/n$ is varying. Results are shown at 3 values of N : 500, 1000, 4000; note steepening of the transition as N increases, in accord with theory. At each unique parameter combination, twenty random problem instances were tried.

parameter vector θ and therefore a good tuning of the algorithm seems necessary. Phase transition provides us a tool for tuning these parameters. The only problem is the dependence of the phase transition on the distribution of nonzero elements of x_o . We represent the value of ρ at the phase transition as $\rho_{\mathcal{A}}^*(\delta, \theta; G)$, where as before G is the distribution of the non-zero elements of x_o . But as mentioned in the last section the distribution of the non-zero elements may not be known beforehand. In order to solve this issue we propose the maximin framework. Suppose that we have a class of distributions \mathcal{G} for the non-zero values of the original vector.

$$\theta^*(\delta) = \arg \max_{\theta} \inf_{G \in \mathcal{G}} \rho^*(\delta, \theta; \mathcal{A}, G) \quad (3.1)$$

In other words for every value of θ the worst distribution is found and the phase transition for that value of parameter and that value of δ is calculated for the least favorable distribution. Then all the phase transition values are compared and the value of θ that exhibits the maximum value of ρ^* is chosen. The optimal phase transition

is called maximin phase transition $\rho^{MM}(\delta, \mathcal{A}; \mathcal{G})$. We may also represent this value by $\rho^{MM}(\delta, \mathcal{A})$ in cases where the family of distributions is clear from the context. In the rest of this chapter we will consider several compressed sensing algorithms and use this framework to tune and compare them. For this purpose an empirical version of the maximin tuning will be used.

3.2 Iterative algorithms

In this note we consider two families of such iterative algorithms.

3.2.1 Simple iterative algorithms

The first family is inspired by the classical relaxation method for approximate solution of large linear systems as explained in Section 2.3. In classical relaxation, one iteratively applies A and its transpose A^* to appropriate vectors and under appropriate conditions, the correct solution is obtained as a limit of the process. While the classical theory is inapplicable to underdetermined systems, it has been found that a sparsity-promoting variant of relaxation can correctly solve such systems, when they have sufficiently sparse solutions. For more information you may refer to Section 2.3 and Section 2.4. Starting from $x^1 = 0$, one repeatedly applies this rule:

$$x^{t+1} = \eta(x^t + \kappa \cdot (A^* r^t); \lambda^t); \quad r^t = y - Ax^t;$$

Here κ is a relaxation parameter ($0 < \kappa < 1$) and we assume throughout that A is normalized so that its columns have unit length. $\eta(\cdot; \lambda)$ denotes a scalar nonlinearity, applied entrywise; we consider both *hard* thresholding $\eta^H(y; \lambda) = y1_{\{|y|>\lambda\}}$ and *soft* thresholding $\eta^S(y; \lambda) = \text{sgn}(y)(|y| - \lambda)_+$. In the above functions λ is called the threshold value. Note that if we set $\eta(y) = y$ we would just have classical relaxation. Iterative Soft Thresholding (IST) with a fixed threshold has been used in various settings more than a decade ago – see for example published work of Sylvain Sardy and co-authors [92]. A formal convergence analysis was given by [28] in the determined case. For more extensive review of the literature on iterative

soft thresholding algorithms and their connection to ℓ_1 minimization you may refer to Chapter 7. Iterative Hard Thresholding (IHT) was reported useful for several underdetermined cases by Starck, Elad, and their co-authors in papers appearing as early as 2004, [96, 97, 52, 50, 14] – often outperforming IST. Other recent examples of such ideas include [65, 58, 11, 94, 13, 12].

These iterative schemes are easy to implement: they require only two matrix-vector products per iteration and some vector additions and subtractions. For certain very large matrices we can rapidly apply A and A^* without representing A as a full matrix – examples include partial Fourier and Hadamard transforms. In such settings, the work required scales very favorably with N (e.g. $N \log(N)$ flops rather than $O(N^2)$).

Actually using such a scheme in practice requires choosing a parameter vector $\theta = (\text{type}, \kappa, \lambda)$; here $\text{type} = S$ or H depending as soft or hard thresholding is required; the other parameters are as earlier. Moreover the threshold value λ needs to vary from iteration to iteration. The general form in which such schemes are often discussed does not give a true ready-to-run algorithm. This keeps potential users from successfully exploiting the idea.

3.2.2 Composite iterative algorithms

In solving determined linear systems, relaxation can often be outperformed by other methods. Because of the similarity of relaxation to IST/IHT schemes, parallel improvements seem worth pursuing in the sparsity setting. A more sophisticated scheme – Two Stage Thresholding (TST) – combines exact solution of small linear systems combined with thresholding before and after this solution. In stage one, we screen for ‘significant’ nonzeros just as in IST and IHT:

$$v^t = \eta^{(1)}(x^t + \kappa A^* r^t; \lambda^{1t}); \quad r^t = y - Ax^t;$$

We let I^t denote the combined support of v^t and x^t and we solve

$$w^t = (A_{I^t}^* A_{I^t})^{-1} A_{I^t}^* y.$$

We then threshold a second time,

$$x^{t+1} = \eta^{(2)}(w^t; \lambda_{2t}),$$

producing a sparse vector. Here the threshold λ and even the nonlinearity $\eta^{(i)}$ might be chosen differently in stages 1 and 2 and might depend on the iteration and on measured signal properties. CoSaMP [80] and subspace pursuit [27] may be considered as special cases of TST. This will be clearer when we explain the threshold choice in the next section.

It seems that the use of explicit solutions to the smaller systems might yield improved performance, although at the cost of potentially much more expense per iteration. An important problem is user reticence. In the case of TST there are even more choices to be made than with IST/IHT. This scheme again presents the ‘recipe ingredients without recipe amounts’ obstacle: users may be turned off by the requirement to specify many such tunable parameters.

3.2.3 Threshold choice

Effective choice of thresholds is a heavily-developed topic in statistical signal processing. We have focused on two families of tunable alternatives.

Interference heuristic. We pretend that the marginal histogram of A^*r at sites in the coefficient vector where $x_o(i) = 0$ is Gaussian, with common standard deviation σ . We robustly estimate the marginal standard deviation σ of the entries in A^*r at a given iteration and set the threshold λ^t as a fixed multiple of that standard deviation – $t = \tau \cdot \sigma^t$, where τ is our chosen threshold control parameter, typically in the range $0 < \tau < 4$. The underlying rationale for this approach is explained in [33] where its correctness was heavily tested. Under this heuristic, we control the threshold τ as in standard detection theory using the False Alarm Rate (FAR); thus $FAR = 2 \cdot \Phi(-\tau)$ where Φ denotes the standard normal distribution function. *Oracle heuristic.* In the TST scheme, imagine that an oracle tells us the true underlying sparsity level k , and we scale the threshold adaptively at each iteration so that at stage 1 we yield $\alpha \cdot k$ nonzeros and at stage two $\beta \cdot k$ nonzeros. The method CoSaMP [80] corresponds to

$\beta = 2$, $\alpha = 1$, while subspace pursuit [27] corresponds to $\beta = \alpha = 1$.

A problem with the oracle heuristic is that, in interesting applications, there is no such oracle, meaning that we wouldn't in practice ever know what k to use. A problem with the interference heuristic is that the Gaussian model may not work when the matrix is not really 'random'.

3.3 Estimating the empirical phase transition

For a given algorithm with a fully specified parameter vector θ , we conduct one phase transition measurement experiment as follows. We fix a *problem suite*, i.e. a matrix ensemble and a coefficient distribution for generating problem instances (A, x_0) and a measure of success; see below. For a fixed $N = 800$, we varied $n = \lceil \delta N \rceil$ and $k = \lceil \rho n \rceil$ through a grid of 900 δ and ρ combinations, with δ varying from .05 to 1 in 30 steps and ρ varying from .05 up to a ceiling value $\rho_{max} < 1$ in as many as 30 steps. We then have a grid of δ, ρ values in parameter space $[0, 1]^2$. At each (δ, ρ) combination, we will take M problem instances and obtain M algorithm outputs \hat{x}_i ; in our case $M = 100$. For each problem instance we declare success if

$$\frac{\|x_o - \hat{x}\|_2}{\|x_o\|_2} \leq \text{tol},$$

where tol is a given parameter, in our case 10^{-2} ; the variable S_i indicates success on the i^{th} Monte Carlo realization. We summarize the M Monte Carlo repetitions by the total number of successes $S = \sum_i S_i$ in those M trials. S is distributed binomial $\text{Bin}(\pi, M)$ where π denotes the success probability $\pi \in [0, 1]$. This probability depends on k, n, N so we write $\pi = \pi(\rho|\delta; N)$.

We *define* the location of the phase transition using logistic regression similarly to [47, 46]. The *finite- N phase transition* is the value of ρ at which success probability crosses 50%:

$$\pi(\rho|\delta; N) = \frac{1}{2} \quad \text{at} \quad \rho = \rho^*(\delta; \theta).$$

This notion is well-known in biometrics where the 50% point of the dose-response is called the LD50. (Actually there is a dependence on the tolerance tol so $\rho^*(\delta; \theta) \equiv$

$\rho^*(\delta; \theta | N, \text{tol})$; but this dependence is found to be weak.)

To *estimate* the phase transition from data, we collect triples $(k, M, S(k, n, N))$ all at one value of (n, N) , and model $S(k, n, N) \sim \text{Bin}(\pi_k; M)$ using a generalized linear model with logistic link

$$\text{logit}(\pi) = a + b\rho,$$

where $\rho = k/n$ and $\text{logit}(\pi) = \log(\frac{\pi}{1-\pi})$; in biometric language, we assume that the dose-response probability follows a logistic curve.

The fitted parameters \hat{a}, \hat{b} , give the estimated phase transition from

$$\hat{\rho}^*(\delta; \theta) = -\hat{a}/\hat{b}.$$

We denote this estimated value by $\rho^*(\delta; \theta)$ in the rest of the paper.

3.4 Tuning procedure

We conducted extensive computational experiments to evaluate the phase transitions of various algorithms. In all, we performed more than 90,000,000 reconstructions, using 38 servers at a commercial dedicated server facility for one month. These calculations would have run more than 3 years on a single desktop computer.

For a fixed iterative scheme and a fixed tuning parameter θ , we considered in turn each of several problem suites $\mathcal{S} = (E, C)$, i.e. several combinations of random matrix ensemble E and coefficient amplitude distributions C . At each such combination, we measured the phase transitions as a function of δ .

In the tuning stage of our project we worked only with the *standard suite* \mathcal{S}_0 ; here the matrix ensemble is the Uniform Spherical Ensemble (USE)¹ matrix and the coefficient ensemble has all nonzeros randomly \pm equiprobable and independent. Below we call this the *CARS* ensemble, short for Constant Amplitude, with Random Signs.

¹The columns of these matrices are iid samples from the uniform distribution on the unit sphere in \mathbb{R}^n .

For a fixed $N = 800$ we made simulations at the standard suite, and measured the empirical phase transition $\rho^*(\delta; \theta)$ as described above. We denote the optimal parameter choice via

$$\theta^*(\delta) = \arg \max_{\theta} \rho^*(\delta; \theta). \quad (3.2)$$

In the later evaluation stage, we will observe that the *CARS* ensemble is the least favorable distribution among the distributions we consider. See Section 3.7.

3.5 Tuning results

Figure 3.3 illustrates tuning results for IST on the standard suite \mathcal{S}_0 . Here $\theta = (\text{RelaxationParameter}, \text{FARParameter})$. Panel (a) shows the different optimized phase transitions available by tuning FAR to depend on δ while the relaxation parameter is fixed. Panel (b) shows the optimally tuned FAR parameters at each given δ and choice of relaxation parameter. Figure 3.4 offers the same information for IHT.

Optimum performance of IST occurs at higher values of the false alarm rate than for IHT. Decreasing the relaxation parameter beyond the range shown here does not improve the results for IST and IHT.

Figure 3.5 illustrates performance of TST for different values of $\theta = (\alpha, \beta)$. Panel (a) shows the different optimized phase transitions available by tuning β at fixed $\alpha = 1$ and Panel (b) shows optimal phase transitions with $\alpha = \beta$ varying. Both displays point to the conclusion that $\alpha = \beta = 1$ dominates other choices. Hence subspace pursuit ($\alpha = 1, \beta = 1$) dominates CoSaMP ($\alpha = 1, \beta = 2$).

3.6 Recommended choices

We provide three versions of iterative algorithms based on our optimal tuning exercise: recommended-IST, recommended-IHT and recommended-TST. They are implemented in Matlab and published at URL

`sparselab.stanford.edu/OptimalTuning/main.htm`. In our recommended versions, there are no free parameters. The user specifies only the matrix A and the left-hand

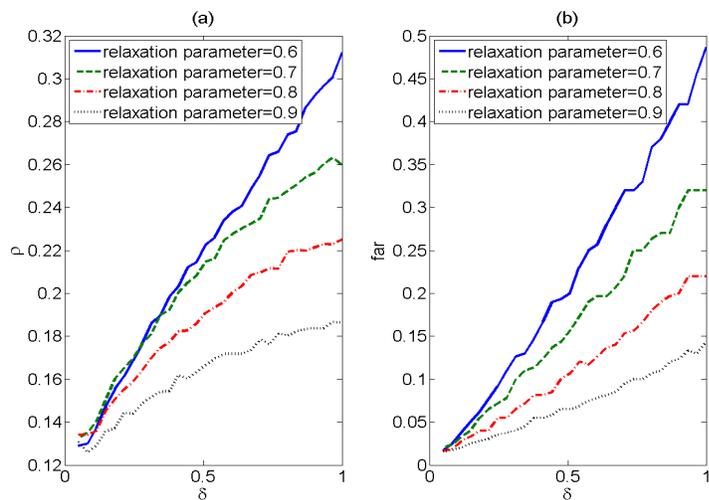


Figure 3.3: (a) Optimized phase transitions for IST at various choices of relaxation parameter κ . (b) FAR parameter choices yielding those optima. The value $\kappa = 0.6$ outperforms the other choices.

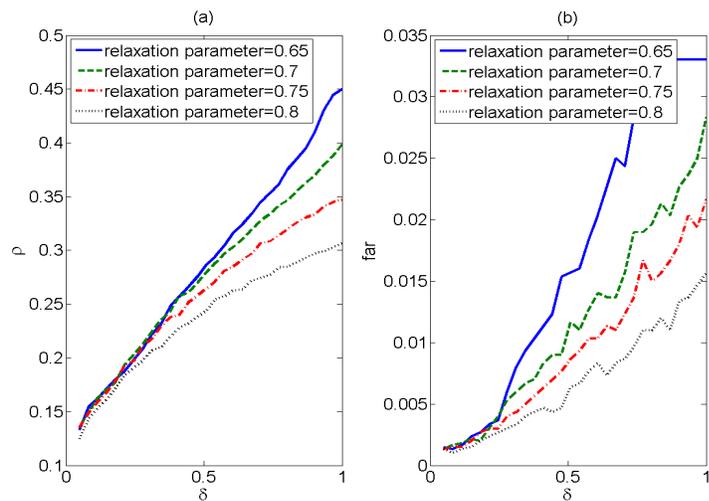


Figure 3.4: (a) Optimized phase transitions for IHT at various choices of relaxation parameter κ . (b) FAR parameter choices yielding those optima. The value $\kappa = 0.6$ outperforms the other choices.

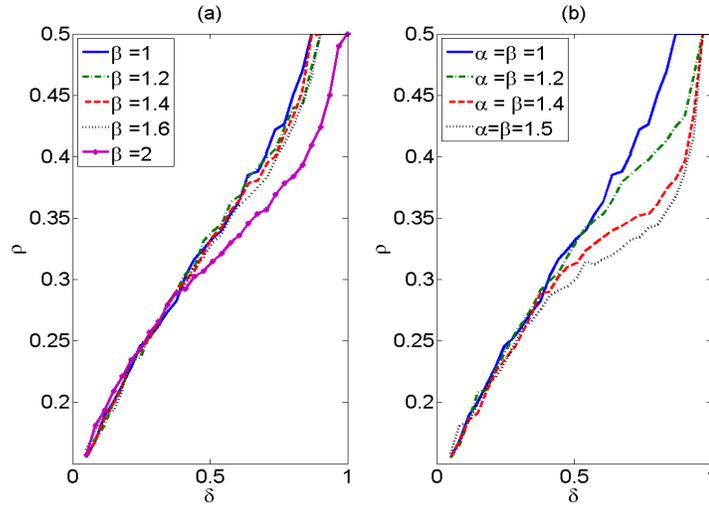


Figure 3.5: (a) Empirical phase transitions of TST- (α, β) for $\alpha = 1$ and different values of β ; $\beta = 1$ performs best; (b) Empirical phase transitions when $\alpha = \beta$; again $\beta = 1$ performs best. In both panels, differences are small for $\delta < .5$, i.e. undersampling factors greater than 2.

side y . In particular the user does not specify the expected sparsity level, which in most applications cannot be considered known.

These recommended algorithms are not the same as previously published algorithms. For example, recommended TST has parameters $\alpha = 1$ and $\beta = 1$, so it initially seems identical to subspace pursuit [27]. However, subspace pursuit demands an *oracle* to inform the user of the true underlying sparsity of the vector. Recommended-TST de facto assumes a specific value for the assumed sparsity level at each given δ (see Table 3.3). If the actual sparsity in x_o is better than the assumed value, the algorithm still works. If the sparsity is actually worse than what Rec-TST assumes, no other tuning of the algorithm will work either. The user does not need to know the assumed sparsity level – it is hard-coded. In effect, we have removed the oracle dependence of the subspace pursuit method.

We remind the reader that these algorithms dominate other implementations in the same class. Thus, recommended TST dominates CoSaMP; this is particularly evident for $\delta > .5$ (see Figure 3.5).

A companion set of algorithms – described later – is available for the case where A is not an explicit matrix but instead a linear operator for which Av and $A'w$ can be computed without storing A as a matrix. Some differences in tuning for that case have been found to be valuable.

We record in the following tables a selection of the optimally tuned parameter values.

Table 3.1: Recommended IST: Optimally tuned FAR parameter and resulting optimized phase transition, both as a function of δ . Recommended value of relaxation parameter $\kappa = 0.6$.

δ	.05	.11	.21	.31	.41	.5	.6	.7	.8	.93
ρ	.124	.13	.16	.18	.2	.22	.23	.25	.27	.29
FAR	.02	.037	.07	.12	.16	.2	.25	.32	.37	.42

Table 3.2: Recommended IHT: Optimally tuned FAR parameter and resulting optimized phase transition, both as a function of δ . Recommended value of relaxation parameter $\kappa = 0.65$.

δ	.05	.11	.21	.41	.5	.6	.7	.8	.93
ρ	.12	.16	.18	.25	.28	.31	.34	.38	.41
$100 \cdot FAR$.15	.2	.4	1.1	1.5	2	2.7	3.5	4.3

Table 3.3: Recommended TST: optimal tuning parameters are $\alpha = \beta = 1$.

δ	.05	.11	.21	.31	.41	.5	.6	.7	.8	.93
ρ	.124	.17	.22	.26	.30	.33	.368	.4	.44	.48

Figure 3.6 compares our recommended implementations with each other and with LARS [49] and OMP [87], as well as, the theoretical phase transition curve for ℓ_1 . The figure depicts empirical phase transitions at the Standard Suite. These transitions

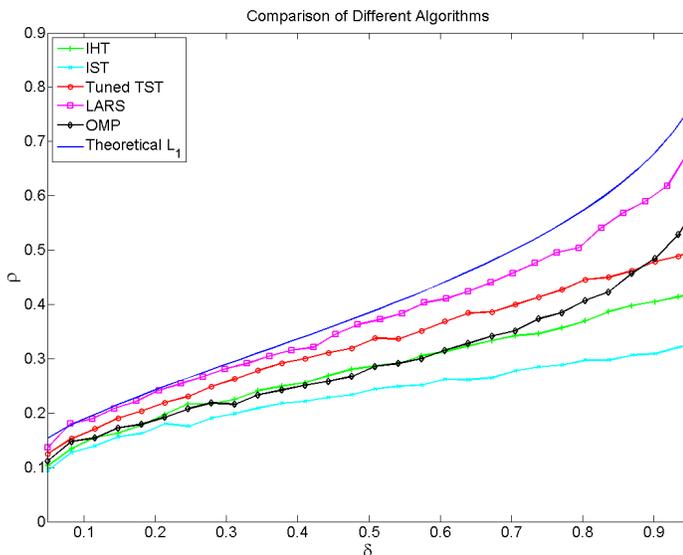


Figure 3.6: Phase Transitions of several algorithms at the standard suite. Upper curve: theoretical phase transition, ℓ_1 minimization; lower curves: Observed transitions of algorithms recommended here.

obey the following ordering:

$$\ell_1 > \text{LARS} > \text{Rec-TST} > \text{Rec-IHT} > \text{Rec-IST},$$

where ℓ_1 refers to the phase transition of the limiting probability for exact reconstruction by ℓ_1 minimization (Figure 3.1), and the other symbols denote empirical transitions of specific algorithms. One might have expected this result based on qualitative grounds; however, it is striking to see how close some of the curves actually are. For example, OMP performance is very similar to tuned IHT for $\delta < 0.7$. These simple iterative algorithms are dramatically easier to program and also dramatically cheaper to run on a per iteration basis, than usual optimization-based approaches. However there is still large gap between the phase transition of iterative thresholding algorithms and the phase transition of ℓ_1 . This issue will be addressed in the next chapter.

3.7 Robustness

A *robust* choice of parameters offers a guaranteed level of performance across all situations. Such a choice can be made by solving the maximin problem as mentioned at the beginning of this chapter

$$\theta^r(\delta) = \arg \max_{\theta} \min_{G \in \mathcal{G}} \rho^*(\delta; \theta; G).$$

In words, we tune θ to give the highest possible performance guarantee valid across a universe \mathcal{G} of ensembles G . The maximin is achieved at the worst case or *least-favorable* distribution; this depends on the given algorithm, the tuning, and the universe of distributions. We considered four coefficient ensembles G : in addition to the CARS ensemble defined above, we considered coefficients from the double exponential distribution, the Cauchy, and the uniform distribution on $[-1, 1]$. As it turns out, our recommended tuning in effect has the maximin property. As described earlier, we tuned at the standard suite \mathcal{S}_0 , with constant amplitude, random-sign (CARS) coefficients and matrices from USE. Figures 3.7-3.8-3.9 display results for Rec-IST, Rec-IHT, and Rec-TST at a range of problem suites. For all three algorithms, the CARS ensemble is approximately the least favorable coefficient ensemble. Since we have tuned at that ensemble, our choice of tuning parameters can be said to be robust. In other words, if the problem suite is different from the standard suite, the phase transition of the algorithm will be even better than the phase transitions tabulated in the tables above. We have also done some experiment to check robustness to the matrix ensemble. Matrix ensembles included the USE defined above, as well as matrices with random \pm entries [Random Sign Ensemble (RSE)] and partial Fourier matrices (to be explained later). Figures 3.10-3.11-3.12 study Rec-IST, Rec-IHT, and Rec-TST at three matrix ensembles: USE, Random Sign Ensemble (RSE) where the elements of the matrix are chosen iid from ± 1 , and Uniform Random Projection (URP) ensemble. Results are similar for the RSE and USE ensembles and usually better for URP. A surprising exception to the above pattern is described below in Section 3.9.

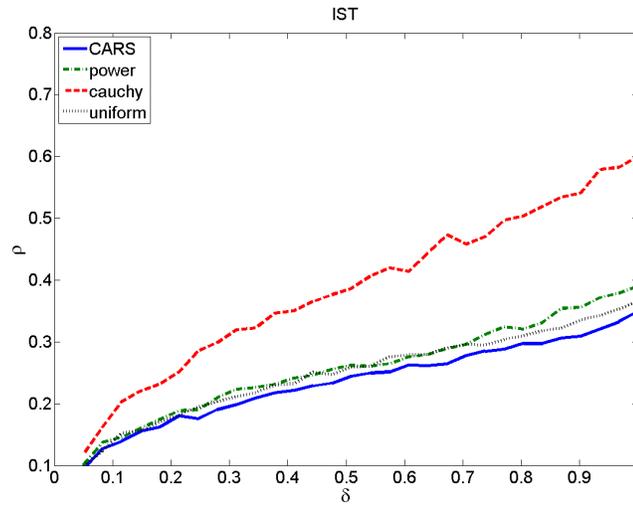


Figure 3.7: Observed phase transitions of recommended IST at different coefficient ensembles.

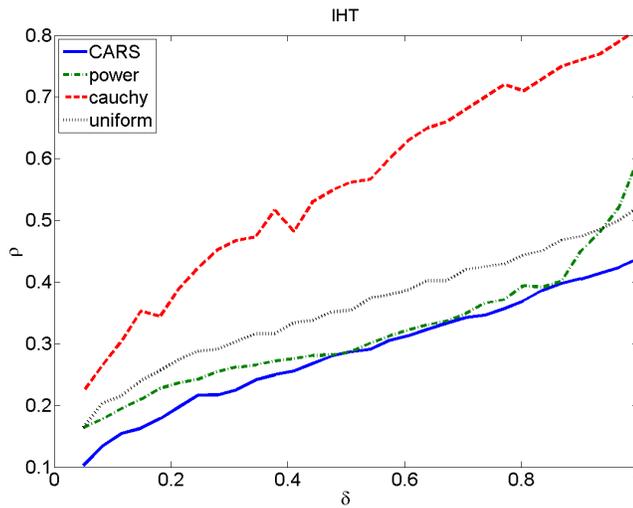


Figure 3.8: Observed phase transition of recommended IHT at different coefficient ensembles.

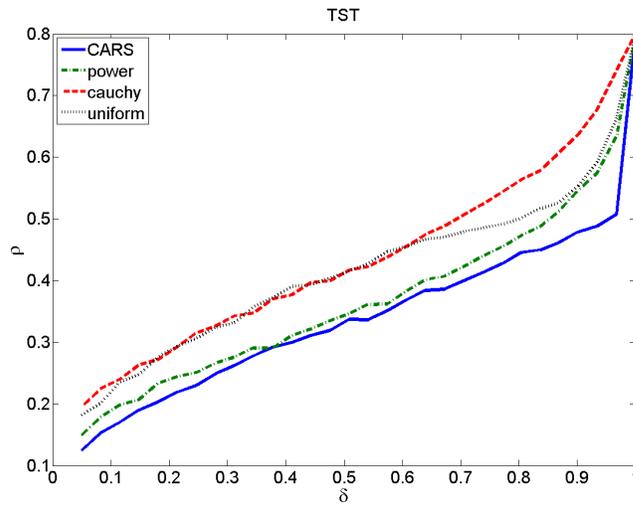


Figure 3.9: Observed phase transition of recommended TST at different coefficient ensembles.

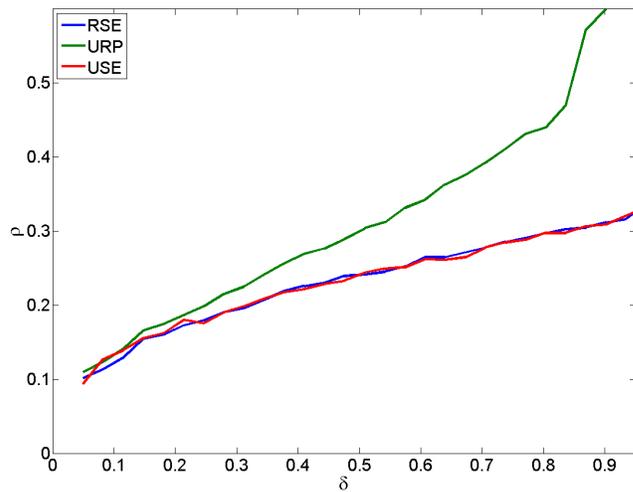


Figure 3.10: Observed phase transition of recommended IST for different matrix ensembles.

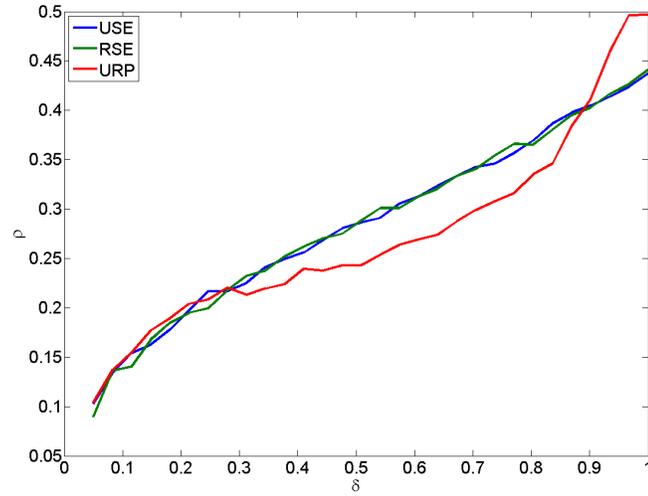


Figure 3.11: Observed phase transition of recommended IHT for different matrix ensembles. Note that the red curve (URP matrix ensemble) is significantly below the blue curve (USE matrix ensemble) for $0.3 < \delta < .85$. This was the only substantial exception observed to the maximin property in our study.

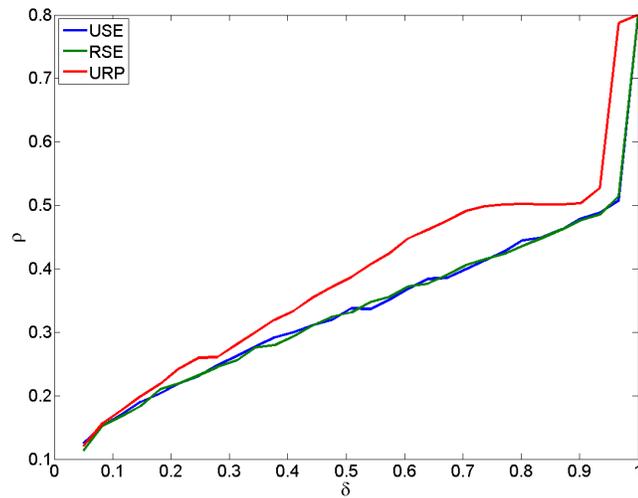


Figure 3.12: Observed phase transition of recommended TST for different matrix ensembles.

3.8 Running times

Algorithm running times are given in Table 3.4. They were measured on an Intel 2 Core Processor (2.13GHz, 4 GBytes RAM). All implementations are in Matlab. In order to give a fair comparison between algorithms with very different internal logic, we ran each iterative algorithm until a convergence criterion was met: $\frac{\|y - A\hat{x}\|_2}{\|y\|_2} \leq .001$.

Table 3.4: Algorithm Timings at the Standard Suite. Average running time (sec) until $\frac{\|y - A\hat{x}\|_2}{\|y\|_2} \leq .001$. Averages cover 10 independent runs. Problem sizes as indicated.

N	δ	ρ	IHT	TST	OMP	LARS
2000	0.9	0.17	10.6	12	20	28
4000	0.9	0.17	44.8	91.2	157	216
6000	0.9	0.17	90	286	537	798
2000	0.7	0.28	7.2	3.3	7.6	11.5
4000	0.7	0.28	28.4	24.5	57.8	98.4
6000	0.7	0.28	64.5	118	188	987
2000	0.5	0.2	5.8	0.91	1.5	2.7
4000	0.5	0.2	23	7	12	20
6000	0.5	0.2	52	23	38	65
8000	0.5	0.2	91	52	97	164
10000	0.5	0.2	130	100	168	270
2000	0.3	0.12	2	0.08	0.25	0.4
4000	0.3	0.12	9	0.65	1.8	2.6
8000	0.3	0.12	34	5	15	22
10000	0.3	0.12	54	13	28.5	38.5

3.9 Ensembles based on fast operators

The matrix ensembles discussed so far all used dense matrices with random elements. However, many applications of sparsity-seeking decompositions use linear operators which are never stored as matrices. Typically such operators can be applied rapidly so we call the resulting measurement operators *FastOps* ensembles. The partial Fourier ensemble [20] provides an example. Here the $n \times N$ matrix A has for its rows a random subset of the N rows in the standard Fourier transform matrix. Av and $A'w$ can both be computed in order $N \log(N)$ time; the comparable dense matrix vector products would cost order N^2 flops. The partial Hadamard ensemble is defined similarly, using the Hadamard matrix of order N in place of the Fourier matrix.

The simple iterative algorithms IHT and IST are particularly suited for use with FastOps ensemble since they require only repetitive application of Av and $A'w$ interleaved with thresholding, and this is exactly how FastOps ensembles are set up to be used.

We considered two FastOps suites: the 1D partial Fourier ensemble and the 1D partial Hadamard ensembles. The *Standard* FastOps suite uses the partial Fourier matrix ensemble and CARS coefficients. Figure 3.13 compares the optimally-tuned performance of IHT, IST and TST at the standard FastOps suite.

We found that

- it is beneficial to tune IHT and IST specially for FastOps ensembles, because the previous tuning (aka maximin tuning) was driven by least favorable cases occurring at non-FastOps ensembles. Here such cases are ruled out, and maximin tuning only considers a narrower range of relevant cases; the achieved maximin phase transition improves.
- For TST, $\alpha = \beta = 1$ is still optimal, but for the maximin tuning restricted to FastOps, ρ^* , improves.
- the relaxation parameter in IHT/IST makes essentially no contribution to performance in this setting.
- 1D partial Hadamard and 1D partial Fourier gave very similar results.

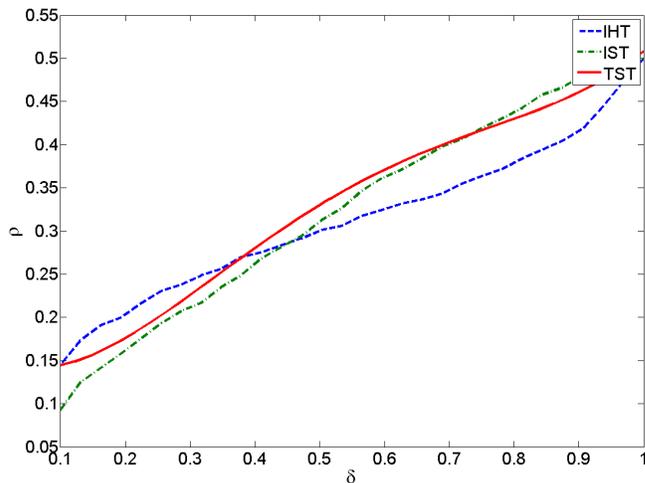


Figure 3.13: Comparison of the performance of recommended IHT, IST and TST for partial Fourier ensemble.

- the performance of IHT is very much in line with earlier results for the random matrix ensembles.
- IST behaves dramatically better at partial Fourier ensembles than for the random matrix ensembles (Figure 3.14) and even outperforms IHT for $\delta > .5$ (Figure 3.13).

Recommended parameters are shown in Tables 3.5-3.6. Running times are studied in Table 3.7. The execution times of both the fast IHT and fast TST scale favorably with the problem size N . In most of our studies TST is faster than IHT and they are both much faster than LARS. The favorable timing results of TST on large problem sizes surprised us.

3.10 Before using these results

Readers may find the following reminders helpful:

- Our software already has embedded within it the appropriate values from the

Table 3.5: Recommended IST, Standard FastOps Suite. Optimally Tuned FAR and optimized phase transition ρ^* of the recommended IST algorithm. Recommended relaxation parameter $\kappa = 1$.

δ	.11	.21	.31	.41	.5	.6	.7	.8	.9
ρ	.092	.16	.21	.26	.31	.37	.41	.44	.48
FAR	.0209	.0736	.13	.19	.26	.32	.32	.32	.32

Table 3.6: Recommended IHT, Standard FastOps Suite. Optimally Tuned FAR and optimized phase transition ρ^* of the recommended IHT algorithm. Recommended relaxation parameter $\kappa = 1$.

δ	.05	.11	.21	.31	.41	.5	.6	.7	.8
ρ	.056	.14	.2	.24	.27	.3	.32	.34	.38
1000FAR	.3	.4	1.8	2.9	3.8	5	5	5	5

tables presented here, so you may not need to copy information from the tables and apply it. However, if you need to code your own implementation, remember that the parameter $\rho = \rho^*$ in our tables specifies the largest workable k^* via $k^* = \lfloor \rho^* \cdot n \rfloor = \lfloor \rho^* \cdot \delta \cdot N \rfloor$.

- Your A matrix must be normalized so that all columns have unit Euclidean norm; for a badly-scaled matrix the algorithms may diverge rapidly.
- The software assumes the sparsity level k is unknown *a priori*, and uses, for each level of the indeterminacy ratio $\delta = n/N$, the largest workable sparsity level k^* . If your application provides an oracle that makes k known in advance, you may wish to customize the code to use this information – but this is not necessary.

Table 3.7: Algorithm Timings at the Standard FastOps Suite. Average running time (sec) until $\frac{\|y - A\hat{x}\|_2}{\|y\|_2} \leq .001$. Averages cover 10 independent runs. Problem sizes as indicated.

N	δ	ρ	IHT	TST	LARS
8192	.1	.1	.5	.1	.6
16384	.1	.1	1.2	.25	2.5
32768	.1	.1	2.56	.48	10.8
65536	.1	.1	8	2.3	65
131072	.1	.1	18	5.6	> 900
262144	.1	.1	39	13	> 900
524288	.1	.1	85	27	> 900
16384	.3	.18	.5	.4	25
8192	.3	.18	.25	.21	5.2
8192	.5	.21	.18	.19	13.5
16384	.5	.21	.38	.4	81

3.11 The computational effort-phase transition trade-off

This project adopted the goal of squeezing the best phase transition performance out of some simple, computationally feasible algorithms. Staring us in the face is the fact that ℓ_1 minimization generally offers better performance (higher phase transitions) than any of our tuned algorithms. We will address this issue in the next chapter. The goal of the next chapter is to design an algorithm as simple as IST and IHT that has the same phase transition curve as ℓ_1 minimization.

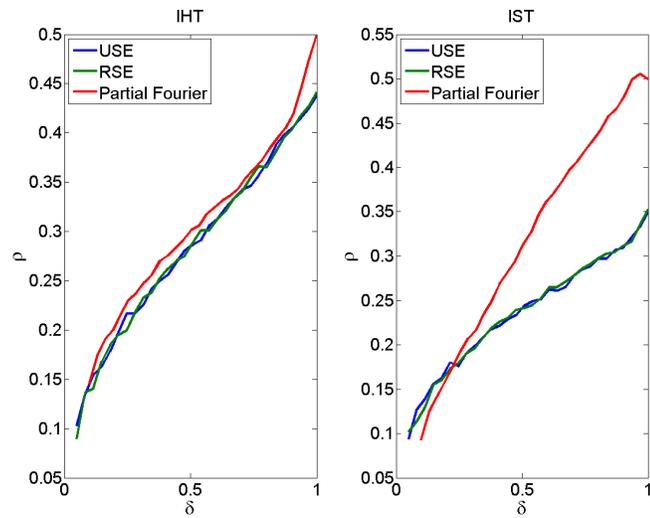


Figure 3.14: (a) Phase transitions of recommended IHT for different matrix ensembles (b) Phase transitions of recommended IST for different matrix ensembles. Generally speaking, the partial Fourier ensemble is not the worst case ensemble; note especially the case *IST* with $\delta > 0.4$.

Chapter 4

Approximate Message Passing

4.1 Introduction

As we showed in the last chapter most of the algorithms that are faster than ℓ_1 perform worse in the sparsity measurement trade-off. In this chapter we develop an iterative algorithm achieving reconstruction performance in one important sense *identical to* ℓ_1 -based reconstruction while running dramatically faster. This algorithm is called Approximate Message Passing or AMP. As before, we assume that a vector y of n measurements is obtained from an unknown N -vector x_0 according to $y = Ax_0$, where A is the $n \times N$ measurement matrix $n < N$. Starting from an initial guess $x^0 = 0$, the *first order approximate message passing* (AMP) algorithm proceeds iteratively according to:

$$x^{t+1} = \eta_t(A^* z^t + x^t), \quad (4.1)$$

$$z^t = y - Ax^t + \frac{1}{\delta} z^{t-1} \langle \eta'_t(A^* z^{t-1} + x^{t-1}) \rangle. \quad (4.2)$$

Here $\eta_t(\cdot)$ are scalar *threshold* functions (applied componentwise), $x^t \in \mathbb{R}^N$ is the current estimate of x_0 , and $z^t \in \mathbb{R}^n$ is the current residual. A^* denotes transpose of A . For a vector $u = (u(1), \dots, u(N))$, $\langle u \rangle \equiv \sum_{i=1}^N u(i)/N$. Finally $\eta'_t(s) = \frac{\partial}{\partial s} \eta_t(s)$.

The iterations of the approximate message passing seem very similar to the iterations of iterative thresholding algorithms which is given by

$$x^{t+1} = \eta_t(A^* z^t + x^t), \quad (4.3)$$

$$z^t = y - Ax^t. \quad (4.4)$$

But Iterative thresholding schemes based on (4.3), (4.4) lack the crucial term in (4.2) – namely, $\frac{1}{\delta} z^{t-1} \langle \eta'_t(A^* z^{t-1} + x^{t-1}) \rangle$ is not included. We derive this term from the theory of belief propagation in graphical models, and show that it substantially improves the sparsity-undersampling tradeoff¹.

Extensive numerical and Monte Carlo work reported here shows that AMP, defined by (4.1), (4.2) achieves a sparsity-undersampling tradeoff matching the theoretical tradeoff which has been proved for LP-based reconstruction. We consider a parameter space with axes quantifying sparsity and undersampling. In the limit of large dimensions N, n , the parameter space splits in two *phases*: one where the AMP approach is successful in accurately reconstructing x_0 and one where it is unsuccessful. References [32, 44, 42] derived regions of success and failure for LP-based recovery. We find these two ostensibly different partitions of the sparsity-undersampling parameter space to be *identical*. Both reconstruction approaches succeed or fail over the same regions, see Figure 4.1.

Our finding has extensive empirical evidence and strong theoretical support. We introduce a *state evolution* formalism and find that it accurately predicts the dynamical behavior of numerous observables of the AMP algorithm. In this formalism, the mean squared error of reconstruction is a state variable; its change from iteration to iteration is modeled by a simple scalar function, the *MSE map*. When this map has nonzero fixed points, the formalism predicts that AMP will not successfully recover the desired solution. The MSE map depends on the underlying sparsity and undersampling ratios, and can develop nonzero fixed points over a region of sparsity/undersampling space. The region is evaluated analytically and found to coincide

¹Here we are considering the class of iterative thresholding algorithms that are using aggressive continuation strategies. There are other iterative thresholding algorithms with exactly the same sparsity measurement tradeoff as ℓ_1 . Those algorithms will be discussed and compared with AMP in Chapter 7.

very precisely (i.e. within numerical precision) with the region over which LP-based methods are proved to fail. Extensive Monte Carlo testing of AMP reconstruction finds the region where AMP fails is, to within statistical precision, the same region.

In short we introduce a fast iterative algorithm which is found to perform as well as corresponding linear programming based methods on random problems. Our findings are supported from simulations and from a theoretical formalism. Remarkably, the success/failure phases of LP reconstruction were previously found by methods in combinatorial geometry; we give here what amounts to a very simple formula for the phase boundary, derived using a very different and seemingly elegant theoretical principle.

4.1.1 Underdetermined linear systems

Let $x_0 \in \mathbb{R}^N$ be the signal of interest. We are interested in reconstructing it from the vector of measurements $y = Ax_0$, with $y \in \mathbb{R}^n$, for $n < N$. For the moment, we assume the entries A_{ij} of the measurement matrix are independent and identically distributed normal $\mathbf{N}(0, 1/n)$.

In this chapter to prove the strength of AMP in dealing with different structures we consider three canonical models for the signal x_0 and three nonlinear reconstruction procedures based on linear programming.

+ : x_0 is nonnegative, with at most k entries different from 0. Reconstruct by solving the LP: minimize $\sum_{i=1}^N x_i$ subject to $x \geq 0$, and $Ax = y$.

\pm : x_0 has as many as k nonzero entries. Reconstruct by solving the minimum ℓ_1 norm problem: minimize $\|x\|_1$, subject to $Ax = y$. This can be cast as an LP.

\square : $x_0 \in [-1, 1]^N$, with at most k entries in the interior $(-1, 1)$. Reconstruction by solving the LP feasibility problem: find any vector $x \in [-1, +1]^N$ with $Ax = y$.

Despite the fact that the systems are underdetermined, under certain conditions on k, n, N these procedures perfectly recover x_0 . This takes place subject to a *sparsity-undersampling tradeoff* namely an upper bound on the signal complexity k relative to n and N .

4.1.2 Phase transitions

We defined the phase transition in Chapter 3 for the problem \pm . The extension of the definition to other problem instances is very simple. However for the sake of self-sufficiency of this chapter we briefly explain it. If we define $\rho = k/n$ and $\delta = n/N$, for each choice of $\chi \in \{+, \pm, \square\}$ there is a function $\rho_{\text{CG}}(\cdot; \chi)$ whose graph partitions the domain into two regions. In the ‘upper’ region, where $\rho > \rho_{\text{CG}}(\delta; \chi)$, the corresponding LP reconstruction $x_1(\chi)$ fails to recover x_0 , in the following sense: as $k, n, N \rightarrow \infty$ in the large system limit with $k/n \rightarrow \rho$ and $n/N \rightarrow \delta$, the probability of exact reconstruction $\{x_1(\chi) = x_0\}$ tends to zero exponentially fast. In the ‘lower’ region, where $\rho < \rho_{\text{CG}}(\delta; \chi)$, LP reconstruction succeeds to recover x_0 , in the following sense: as $k, n, N \rightarrow \infty$ in the large system limit with $k/n \rightarrow \rho$ and $n/N \rightarrow \delta$, the probability of exact reconstruction $\{x_1(\chi) = x_0\}$ tends to one exponentially fast. We refer to [32, 44, 45, 42] for proofs and precise definitions of the curves $\rho_{\text{CG}}(\cdot; \chi)$.

The three functions $\rho_{\text{CG}}(\cdot; +)$, $\rho_{\text{CG}}(\cdot; \pm)$, $\rho_{\text{CG}}(\cdot; \square)$ are shown in Figure 4.1; they are the red, blue, and green curves, respectively. The ordering $\rho_{\text{CG}}(\delta; +) > \rho_{\text{CG}}(\delta; \pm)$ (red $>$ blue) says that knowing that a signal is sparse and positive is more valuable than only knowing it is sparse. Both the red and blue curves behave as $\rho_{\text{CG}}(\delta; +, \pm) \sim (2 \log(1/\delta))^{-1}$ as $\delta \rightarrow 0$; surprisingly large amounts of undersampling are possible, if sufficient sparsity is present. In contrast, $\rho_{\text{CG}}(\delta; \square) = 0$ (green curve) for $\delta < 1/2$ so the bounds $[-1, 1]$ are really of no help unless we use a limited amount of undersampling, i.e. by less than a factor of two.

4.2 Statistical heuristic for iterative approaches

In Chapter 2 we discussed a linear algebraic heuristic that led us to iterative thresholding algorithms. Since we are using the statistical properties of the problem in AMP, it is important to understand the statistical heuristic for iterative approaches. We will later use this heuristic for AMP. The case \pm has been most discussed and we focus on that case for this section. Imagine first of all that A is an orthogonal matrix,

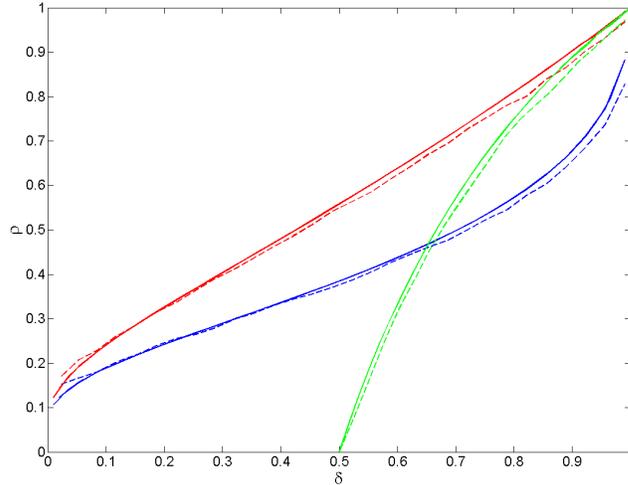


Figure 4.1: The phase transition lines for reconstructing sparse non-negative vectors (problem $+$, red), sparse signed vectors (problem \pm , blue) and vectors with entries in $[-1, 1]$ (problem \square , green). Continuous lines refer to analytical predictions from combinatorial geometry or the state evolution formalisms. Dashed lines present data from experiments with the AMP algorithm, with signal length $N = 1000$ and $T = 1000$ iterations. For each value of δ , we considered a grid of ρ values, at each value, generating 50 random problems. The dashed line presents the estimated 50th percentile of the response curve. At that percentile, the root mean square error after T iterations obeys $\sigma_T \leq 10^{-3}$ in half of the simulated reconstructions.

in particular $A^* = A^{-1}$. Then the iteration (4.1)-(4.2) stops in 1 step, correctly finding x_0 . Next, imagine that A is an invertible matrix; [28], has shown that a related thresholding algorithm with clever scaling of A^* and clever choice of threshold, will correctly find x_0 . Of course both of these motivational observations assume $n = N$, so we are not really *undersampling*.

We sketch a motivational argument for thresholding in the truly undersampled case $n < N$ which is statistical, which has been popular with engineers [74] and which leads to a proper ‘psychology’ for understanding our results. Consider the operator $H = A^*A - I$, and note that $A^*y = x_0 + Hx_0$. If A were orthogonal, we would of course have $H = 0$, and the iteration would, as we have seen immediately succeed in one step. If A is a Gaussian random matrix and $n < N$, then of course A is

not invertible and A^* is not A^{-1} . Instead of $Hx_0 = 0$, in the undersampled case Hx_0 behaves as a kind of noisy random vector, i.e. $A^*y = x_0 + \text{noise}$. Now x_0 is supposed to be a sparse vector, and, one can see, the **noise** term is accurately modeled as a vector with iid Gaussian entries with variance $n^{-1}\|x_0\|_2^2$. This is summarized in the following lemma.

Lemma 4.2.1. *Let A_{ij} be the ij^{th} element of the measurement matrix and suppose that these elements are drawn iid from a given distribution with the following properties, $\mathbb{E}(A_{ij}) = 0$, $\mathbb{E}(A_{ij}^2) = \frac{1}{n}$ and $\mathbb{E}(A_{ij}^4) = O(\frac{1}{n^2})$. We assume that the ratio $\delta = \frac{n}{N}$ is fixed. Let the vector x be a vector with elements iid from a different distribution and independent of the matrix A with a bounded second moment. If $s = (A^*A - I)x$, then*

$$\begin{aligned}\mathbb{E}(s_i) &= 0, \\ \mathbb{E}(s_i s_j) &= O\left(\frac{1}{n}\right), \\ \mathbb{E}(s_i^2) &= \frac{\mathbb{E}(x_i^2)}{\delta} + O\left(\frac{1}{n}\right).\end{aligned}$$

The proof of this lemma is summarized in Appendix E.

In short, the first iteration gives us a ‘noisy’ version of the sparse vector we are seeking to recover. The problem of recovering a sparse vector from noisy measurements has been heavily discussed [36], [67]. and it is well understood that soft thresholding can produce a reduction in mean-squared error when sufficient sparsity is present and the threshold is chosen appropriately. Consequently, one anticipates that x^1 will be closer to x_0 than A^*y .

At the second iteration, one has $A^*(y - Ax^1) = x_0 + H(x_0 - x^1)$. Naively, the matrix H does not correlate with x_0 or x^1 , and so we might pretend that $H(x_0 - x^1)$ is again a Gaussian vector whose entries have variance $n^{-1}\|x_0 - x^1\|_2^2$. This ‘noise level’ is smaller than at iteration zero, and so thresholding of this noise can be anticipated to produce an even more accurate result at iteration two; and so on.

There is a valuable digital communications interpretation of this process. The vector $w = Hx_0$ is the cross-channel interference or *mutual access interference* (MAI),

i.e. the noiselike disturbance each coordinate of A^*y experiences from the *presence* of all the other ‘weakly interacting’ coordinates. The thresholding iteration suppresses this interference in the sparse case by detecting the many ‘silent’ channels and setting them a priori to zero, producing a putatively better guess at the next iteration. At that iteration, the remaining interference is proportional not to the size of the estimand, but instead to the estimation error, i.e. it is caused by the *errors* in reconstructing all the weakly interacting coordinates; these errors are only a fraction of the sizes of the estimands and so the error is significantly reduced at the next iteration.

4.2.1 State evolution

The above ‘sparse denoising’/‘interference suppression’ heuristic, does agree qualitatively with the actual behavior one can observe in sample reconstructions. It is very tempting to take it literally. Assuming it is literally true that the MAI is Gaussian and independent from iteration to iteration, we can formally track the evolution, from iteration to iteration, of the mean-squared error.

This gives a recursive equation for the *formal MSE*, i.e. the MSE which would be true if the heuristic were true. This takes the form

$$\sigma_{t+1}^2 = \Psi(\sigma_t^2), \quad (4.5)$$

$$\Psi(\sigma^2) \equiv \mathbb{E}\left\{\left[\eta\left(X + \frac{\sigma}{\sqrt{\delta}}Z; \lambda\sigma\right) - X\right]^2\right\}. \quad (4.6)$$

Here expectation is with respect to independent random variables $Z \sim \mathbf{N}(0, 1)$ and X , whose distribution coincides with the empirical distribution of the entries of x_0 . We use soft thresholding if the signal is sparse and signed, i.e. if $\chi = \pm$. In the case of sparse non-negative vectors, $\chi = +$, we will let $\eta(u; \lambda\sigma, +) = \max(u - \lambda\sigma, 0)$. Finally, for $\chi = \square$, we let $\eta(u; \square) = \text{sign}(u) \min(|u|, 1)$. Calculations of this sort are familiar from the theory of soft thresholding of sparse signals [67]. We call $\Psi : \sigma^2 \mapsto \Psi(\sigma^2)$ the *MSE map*.

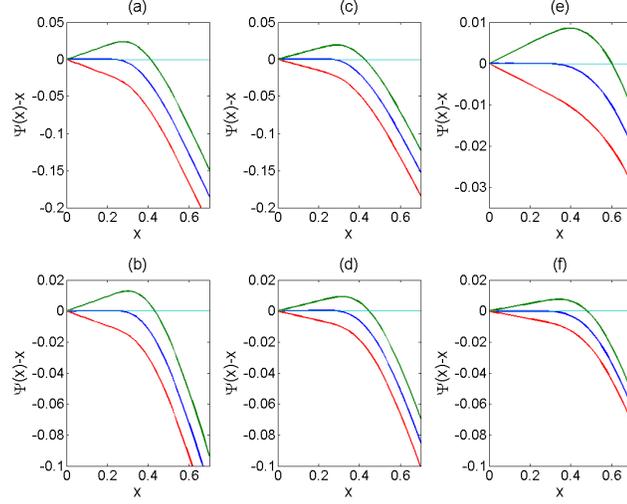


Figure 4.2: Development of fixed points for formal MSE evolution. Here we plot $\Psi(\sigma^2) - \sigma^2$ where $\Psi(\cdot)$ is the MSE map for $\chi = +$ (left column), $\chi = \pm$ (center column) and $\chi = \square$ (right column), $\delta = 0.1$ (upper row, $\chi \in \{+, \pm\}$), $\delta = 0.55$ (upper row, $\chi = \square$), $\delta = 0.4$ (lower row, $\chi \in \{+, \pm\}$) and $\delta = 0.75$ (lower row, $\chi = \square$). A crossing of the y-axis corresponds to a fixed point of Ψ . If the graphed quantity is negative for positive σ^2 , Ψ has no fixed points for $\sigma > 0$. Different curves correspond to different values of ρ : where ρ is respectively less than, equal to and greater than ρ_{SE} . In each case, Ψ has a stable fixed fixed point at zero for $\rho < \rho_{SE}$, and no other fixed points, an unstable fixed point at zero for $\rho = \rho_{SE}$ and develops two fixed points at $\rho > \rho_{SE}$. Blue curves correspond to $\rho = \rho_{SE}(\delta; \chi)$, green to $\rho = 1.05 \cdot \rho_{SE}(\delta; \chi)$, red to $\rho = 0.95 \cdot \rho_{SE}(\delta; \chi)$.

Definition 4.2.1. Given implicit parameters $(\chi, \delta, \rho, \lambda, F)$, with $F = F_X$ the distribution of the random variable X . State Evolution is the recursive map (one-dimensional dynamical system): $\sigma_t^2 \mapsto \Psi(\sigma_t^2)$.

Implicit parameters $(\chi, \delta, \rho, \lambda, F)$ stay fixed during the evolution. Equivalently, the full state evolves by the rule

$$(\sigma_t^2; \chi, \delta, \rho, \lambda, F_X) \mapsto (\Psi(\sigma_t^2); \chi, \delta, \rho, \lambda, F_X).$$

The parameter space is partitioned into two regions:

Region (I): $\Psi(\sigma^2) < \sigma^2$ for all $\sigma^2 \in (0, \mathbb{E}X^2]$. Here $\sigma_t^2 \rightarrow 0$ as $t \rightarrow \infty$: the SE

converges to zero.

Region (II): The complement of Region (I). Here, the SE recursion does *not* evolve to $\sigma^2 = 0$.

The partitioning of parameter space induces a notion of sparsity threshold, the minimal sparsity guarantee needed to obtain convergence of the formal MSE:

$$\rho_{\text{SE}}(\delta; \chi, \lambda, F_X) \equiv \sup \{ \rho : (\delta, \rho, \lambda, F_X) \in \text{Region (I)} \} . \quad (4.7)$$

The subscript _{SE} stands for State Evolution. Of course, ρ_{SE} depends on the case $\chi \in \{+, \pm, \square\}$; it also seems to depend also on the signal distribution F_X ; however, an essential simplification is provided by

Proposition 4.2.2. *For the three canonical problems $\chi \in \{+, \pm, \square\}$, any $\delta \in [0, 1]$, and any random variable X with the prescribed sparsity and bounded second moment, $\rho_{\text{SE}}(\delta; \chi, \lambda, F_X)$ is independent of F_X .*

Independence from F allows us to write $\rho_{\text{SE}}(\delta; \chi, \lambda)$ for the sparsity thresholds. The proof of this statement is sketched in Appendix E, along with the derivation of a more explicit expression.

4.3 Message passing algorithm

Unhappily the formal MSE does not predict the properties of iterative thresholding algorithms. Numerical simulations show very clearly that the MSE map does not describe the evolution of the actual MSE under iterative thresholding. The mathematical reason for this failure is quite simple. After the first iteration, the entries of x^t become strongly dependent, and State Evolution does not predict the moments of x^t . The main surprise of this chapter is that this failure is not the end of the story. We now consider a modification of iterative thresholding inspired by message passing algorithms for inference in graphical models [88], and graph-based error correcting codes [60, 90]. These are iterative algorithms, whose basic variables (‘messages’) are associated to directed edges in a graph that encodes the structure of the statistical

model. The relevant graph here is a complete bipartite graph over N nodes on one side ('variable nodes'), and n on the others ('measurement nodes'). Messages are updated according to the rules

$$x_{i \rightarrow a}^{t+1} = \eta_t \left(\sum_{b \in [n] \setminus a} A_{bi} z_{b \rightarrow i}^t \right), \quad (4.8)$$

$$z_{a \rightarrow i}^t = y_a - \sum_{j \in [p] \setminus i} A_{aj} x_{j \rightarrow a}^t, \quad (4.9)$$

for each $(i, a) \in [N] \times [n]$. We will refer to this algorithm² as MP. We will explain a framework for deriving these algorithms in the next chapter. The main goal of this chapter is to derive the properties of this new algorithm on the compressed sensing problems.

MP has one important drawback with respect to iterative thresholding. Instead of updating N estimates, at each iterations we need to update Nn messages, thus increasing significantly the algorithm complexity. On the other hand, it is easy to see that the right-hand side of (4.8) depends weakly on the index a (only one out of n terms is excluded) and that the right-hand side of (4.9) depends weakly on i . Neglecting altogether this dependence leads to the iterative thresholding (4.3), (4.4). A more careful analysis of this dependence leads to corrections of order one in the high-dimensional limit. Such corrections are however fully captured by the last term on the right hand side of (4.2), thus leading to the AMP algorithm. Statistical physicists would call this the 'Onsager reaction term'; see [102]. In order to justify this we assume that the messages can be approximated in the following way.

$$\begin{aligned} x_{i \rightarrow a}^t &= x_i^t + \delta x_{i \rightarrow a}^t + O(1/N), \\ z_{a \rightarrow i}^t &= z_a^t + \delta z_{a \rightarrow i}^t + O(1/N), \end{aligned} \quad (4.10)$$

with $\delta x_{i \rightarrow a}^t, \delta z_{a \rightarrow i}^t = O(\frac{1}{\sqrt{N}})$ (here the $O(\cdot)$ errors are uniform in the choice of the

²For earlier applications of MP to compressed sensing see [72, 93, 116]. Relations between MP and LP were explored in a number of papers, see for instance [109, 5], albeit from a different perspective.

edge). We also consider a general message passing algorithms of the form

$$x_{i \rightarrow a}^{t+1} = \eta_t \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t \right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (4.11)$$

with $\{\eta_t(\cdot)\}_{t \in \mathbb{N}}$ a sequence of differentiable nonlinear functions with bounded derivatives. In the interest of simplicity, we shall stick to the differentiable model.

Lemma 4.3.1. *Suppose that the asymptotic behavior (4.10) holds for the message passing algorithm (4.8) and (4.9). Then x_i^t and z_a^t satisfy the following equations*

$$x_i^{t+1} = \eta_t \left(\sum_a A_{ia} z_a^t + x_i^t \right) + o_N(1), \quad (4.12)$$

$$z_a^t = y_a - \sum_j A_{aj} x_j^t + \frac{1}{\delta} z_a^{t-1} \langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1}) \rangle + o_N(1), \quad (4.13)$$

where the $o_N(1)$ terms vanish as $N, n \rightarrow \infty$.

The proof of this lemma may be found in Appendix E.

Although AMP seems very similar to simple iterative thresholding (4.3)-(4.4), SE accurately describes its properties, but not those of the standard iteration.

We have conducted extensive simulation experiments with AMP, and more limited experiments with MP, which is computationally more intensive (for details see the simulation section). These experiments show that the performance of the algorithms can be accurately modeled using the MSE map. Let's be more specific.

According to SE, performance of the AMP algorithm is predicted by tracking the evolution of the formal MSE σ_t^2 via the recursion (4.5). Although this formalism is quite simple, it is accurate in the high dimensional limit. Corresponding to the formal quantities calculated by SE are the actual quantities, so of course to the formal MSE corresponds the true MSE $N^{-1} \|x^t - x_0\|_2^2$. Other quantities can be computed in terms of the state σ_t^2 as well: for instance the true false alarm rate $(N - k)^{-1} \#\{i : x^t(i) \neq 0 \text{ and } x_0(i) = 0\}$ is predicted via the formal false alarm rate $\mathbb{P}\{\eta_t(X + \delta^{-1/2} \sigma_t Z) \neq 0 | X = 0\}$. Analogously, the true missed-detection rate $k^{-1} \#\{i : x^t(i) = 0 \text{ and } x_0(i) \neq 0\}$ is predicted by the formal missed-detection rate

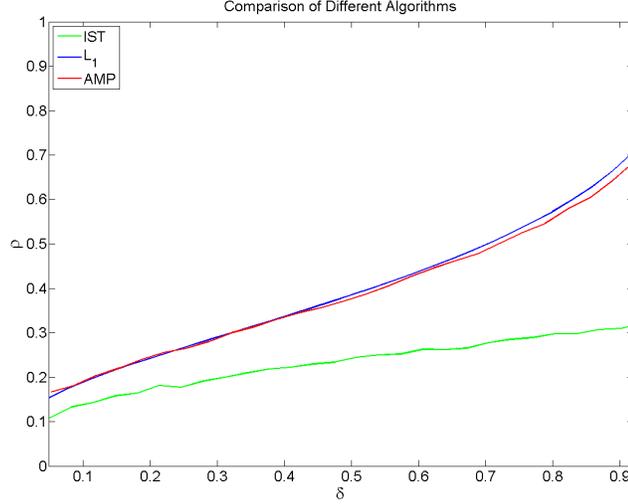


Figure 4.3: Observed phase transitions of reconstruction algorithms. Algorithms studied include iterative soft and hard thresholding, orthogonal matching pursuit, and related. Parameters of each algorithm are tuned to achieve the best possible phase transition [76]. Reconstructions signal length $N = 1000$. Iterative thresholding algorithms used $T = 1000$ iterations. Phase transition curve displays the value of $\rho = k/n$ at which success rate is 50%.

$\mathbb{P}\{\eta_t(X + \delta^{-1/2}\sigma_t Z) = 0 | X \neq 0\}$, and so on.

Our experiments establish agreement of actual and formal quantities.

Finding 1. *For the AMP algorithm, and large dimensions N, n , we observe*

I. *SE correctly predicts the evolution of numerous statistical properties of x^t with the iteration number t . The MSE, the number of nonzeros in x^t , the number of false alarms, the number of missed detections, and several other measures all evolve in way that matches the state evolution formalism to within experimental accuracy.*

II. *SE correctly predicts the success/failure to converge to the correct result. In particular, SE predicts no convergence when $\rho > \rho_{SE}(\delta; \chi, \lambda)$, and convergence if $\rho < \rho_{SE}(\delta; \chi, \lambda)$. This is indeed observed empirically.*

Analogous observations were made for MP.

4.3.1 Optimizing the MP phase transition

In the previous chapter we observed that if we use a version of an algorithm which is not tuned properly it will perform very poorly. This is true about AMP as well. An inappropriately tuned version of MP/AMP will not perform well compared to other algorithms, for example LP-based reconstructions. However, SE provides a natural strategy to tune MP and AMP theoretically. This section is devoted to this derivation.

Maximin tuning

Although the maximin tuning was explained in great detail in the previous chapter, for the sake of completeness of this chapter we summarize our discussion here. Consider a sparse recovery algorithm \mathcal{A}_θ . θ includes all the free parameters of the algorithm. The goal of the maximin tuning is to tune the parameters so that the algorithm achieves its highest phase transition over a set of distributions. Suppose that we have a class of distributions \mathcal{G} for the non-zero values of the original vector.

$$\theta^*(\delta) = \arg \max_{\theta} \inf_{G \in \mathcal{G}} \rho^*(\delta, \theta; \mathcal{A}, G) \quad (4.14)$$

In other words for every value of θ the worst distribution is found and the phase transition for that value of parameter and that value of δ is calculated for the least favorable distribution. Then all the phase transition values are compared and the value of θ that exhibits the maximum value of ρ^* is chosen. In the next section we discuss a theoretical framework that aims to predict the performance of the approximate message passing algorithm.

Maximin tuning of AMP

In the AMP algorithm that uses $\lambda\sigma$ thresholding policy, the only parameter that shall be set according to the maximin framework is λ . Adopt the notation

$$\rho_{SE}(\delta, \chi) = \sup_{\lambda \geq 0} \rho_{SE}(\delta; \chi, \lambda). \quad (4.15)$$

It is easy to see that this supremum is achieved and we denote this value of λ by $\lambda_\chi(\delta)$, $\chi \in \{+, \pm, \square\}$, and refer to the resulting algorithms as to *Maximin tuned MP/AMP* (or sometimes MPM/AMPM for short). The optimal value of λ for $\chi \in \{\pm, +\}$ is given by the following formula

$$\lambda_\chi(\delta) = \frac{1}{\sqrt{\delta}} \arg \max_{z \geq 0} \left\{ \frac{1 - (\kappa_\chi/\delta) [(1+z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - \kappa_\chi [(1+z^2)\Phi(-z) - z\phi(z)]} \right\},$$

where $\kappa_\chi = 1$ for $\chi = +$ and is equal to two for $\chi = \pm$. This formula is drawn in the proof of Proposition 4.2.2 in Appendix E.

High precision numerical evaluations of such expression uncovers the following very suggestive

Finding 2. *For the three canonical problems $\chi \in \{+, \pm, \square\}$, and for any $\delta \in (0, 1)$*

$$\rho_{SE}(\delta; \chi) = \rho_{CG}(\delta; \chi). \quad (4.16)$$

In short, the formal MSE evolves to zero exactly over the same region of (δ, ρ) phase space as does the phase diagram for the corresponding convex optimization! Not only the algorithm converges to the correct solution on the same region as LP does, but also the convergence rate is exponentially fast. The following theorem formalizes this statement.

Theorem 4.3.1. *For $\delta \in [0, 1]$, $\rho < \rho_{SE}(\delta; \chi)$, and any associated random variable X , the formal MSE of optimally-tuned AMP/MP evolves to zero under SE. Viceversa, if $\rho > \rho_{SE}(\delta; \chi)$, the formal MSE does not evolve to zero. Further, for $\rho < \rho_{SE}(\delta; \chi)$, there exists $b = b(\delta, \rho) > 0$ with the following property. If σ_t^2 denotes the formal MSE*

after t SE steps, then, for all $t \geq 0$

$$\sigma_t^2 \leq \sigma_0^2 \exp(-bt). \quad (4.17)$$

The proof of this theorem is summarized in Appendix E.

4.4 Other thresholding policies

So far we have considered three different canonical models and discussed the corresponding AMP algorithms. Hereafter, we focus on just AMP for the canonical problem \pm . For this problem we have considered a very special form of the thresholding policy; the threshold is proportional to the standard deviation of MAI noise. This section aims to show that this thresholding policy is “optimal”. Consider a thresholding policy that can be a “general” function of σ , in contrast to the linear function that was considered in the previous section. As before the state evolution is defined as

$$\sigma_t^2 \mapsto \Psi(\sigma_t^2) = \frac{1}{\delta} \mathbb{E}\{[\eta(X + \sigma_t Z; \lambda(\sigma_t)) - X]^2\}. \quad (4.18)$$

Suppose that the state evolution explains the performance of an algorithm. The thresholding policy $\lambda(\sigma)$ may be considered as a free parameter in this algorithm. Clearly, this parameter affects the phase transition of the algorithm and it is therefore important to tune it optimally. The goal of this section is to find the maximin threshold policy for the iterative algorithm that satisfies SE. Consider the class of distributions $\mathcal{F}_{\epsilon, \gamma} = \{F(\mu) : F(0+) - F(0-) > 1 - \epsilon, E_F(\mu^2) \leq \epsilon\gamma^2\}$. Whenever we remove the parameter γ from the subscript it means that we are considering the set of all distributions that have a mass of at least $1 - \epsilon$ at 0 without any second moment constraint. We also use the notation $F_{\epsilon, \gamma}(\mu) = (1 - \epsilon)\delta_0 + \epsilon\delta_\gamma$ for the distribution that puts a mass of $1 - \epsilon$ at 0 and a mass of ϵ at γ . Clearly, $F_{\epsilon, \gamma}(\mu) \in \mathcal{F}_{\epsilon, \gamma}$. Finally, $\mathcal{G}_\gamma = \{G : E_G\mu^2 \leq \gamma^2\}$. G_γ includes the set of distributions with the second moment bounded above by γ^2 .

Definition 4.4.1. *The risk of the soft thresholding function is defined as,*

$$r(\mu, \lambda; \sigma) = E(\eta(\mu + \sigma Z; \lambda) - \mu)^2, \quad (4.19)$$

where $Z \sim N(0, 1)$ and η is the soft thresholding function as defined before.

Lemma 4.4.1. *$r(\mu, \lambda; \sigma)$ is a concave function of μ^2 .*

Proof. To prove the concavity, we use the second order conditions. For notational simplicity we assume that $\mu > 0$. We have

$$\begin{aligned} \frac{d}{d\mu^2} r(\mu, \lambda; \sigma) &= \frac{1}{\mu} E[-I(-\lambda < \mu + \sigma Z < \lambda)](\eta(\mu + \sigma Z; \lambda) - \mu) \\ &= E[I(-\lambda < \mu + \sigma Z < \lambda)] = \int_{-\frac{\lambda-\mu}{\sigma}}^{\frac{\lambda-\mu}{\sigma}} \phi_Z(z) dz. \end{aligned}$$

Therefore, the second derivative is given by

$$\frac{d^2}{d^2\mu^2} r(\mu, \lambda; \sigma) = -\frac{1}{\sigma\mu} \left(\phi\left(\frac{\lambda-\mu}{\sigma}\right) + \phi\left(\frac{\lambda+\mu}{\sigma}\right) \right).$$

Since $\lambda, \mu > 0$, the second derivative is negative and therefore the risk function is concave. \square

Definition 4.4.2. *Minimum risk thresholding policy is defined as*

$$\lambda_{MR}^*(\sigma; \epsilon, \gamma) = \arg \min_{\lambda} \sup_{F \in \mathcal{F}_{\epsilon, \gamma}} E_{\mu \sim F} r(\mu, \lambda; \sigma). \quad (4.20)$$

Theorem 4.4.2. *Under the minimum risk thresholding policy $\lambda_{MR}^*(\sigma; \epsilon, \gamma)$ the SE phase transition happens at:*

$$\rho_{SE}(\delta, \lambda_{MR}^*) = \max_z \left(\frac{1 - 2/\delta[(1+z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - 2[(1+z^2)\Phi(-z) - z\phi(z)]} \right). \quad (4.21)$$

Furthermore,

$$\rho_{SE}(\delta, \lambda_{MR}^*) = \sup_{\lambda(\sigma)} \inf_{G \in \mathcal{G}_{\gamma}} \rho_{SE}(\delta, \lambda(\sigma), G). \quad (4.22)$$

Although the above theorem claims that the minimum risk thresholding policy is optimal in the maximin sense, it should be emphasized that it is not a legitimate thresholding policy yet, since in the maximin framework the sparsity level of the signal is not known beforehand, while the minimum risk thresholding policy is using this information which is hidden in the parameter ϵ . To fix this issue, we consider the minimum risk thresholding policy and plug in the value of the $\rho_{SE}(\delta, \lambda_{MR}^*)$. From the proof of the above theorem it is clear that this new thresholding policy that does not use the sparsity level still exhibits phase transition at exactly the same place as $\lambda_{MR}^*(\sigma; \epsilon, \gamma)$. The proof of the theorem is moved to the appendix. Here we just mention parts of the proof that explain several interesting properties of the algorithm.

Corollary 4.4.3. *The least favorable distribution that achieves the supremum in $\sup_{F \in \mathcal{F}_{\epsilon, \gamma}} E_{\mu \sim FT}(\mu, \lambda; \sigma)$ is independent of the standard deviation of the noise σ .*

Corollary 4.4.4. *For any thresholding policy $\lambda(\sigma)$, and $G \in \mathcal{F}_{\epsilon, \gamma}$ we have*

$$\Psi(\sigma^2; G) \leq \Psi(\sigma^2; F_{\epsilon, \gamma}), \quad \forall \sigma. \quad (4.23)$$

In other words a two point prior is the least favorable prior for the algorithm.

It should be mentioned that the least favorable prior is not unique. All the distributions that have a point mass of $1 - \epsilon$ at zero and mass of ϵ on the set $\{-\gamma, \gamma\}$ are least favorable as well. Although it is possible to calculate the minimum risk thresholding policy as a function of σ, δ , there are a few obstacles in using such a policy. First there is no explicit solution for the optimal λ in terms of σ and δ and therefore, we have to save a huge table in a memory. Second, we may need an oracle information about γ . There are other simpler thresholding policies that can achieve the same phase transition.

Fixing false alarm rate

In the first three sections of this chapter we considered a class of thresholding policies in which $\lambda(\sigma) = \beta\sigma$ where β just depends on δ . This is called the fixed false alarm rate thresholding policy. A false alarm happens when an element whose actual value

is zero passes the threshold due to noise. By considering this thresholding policy the number of free parameters reduce to one. According to the maximin framework, the optimal value of this parameter is given by

$$\beta^*(\delta) = \arg \max_{\beta \in [0, \infty)} \inf_{G \in \mathcal{G}_\gamma} \rho(\delta, \beta, G). \quad (4.24)$$

The following theorem is shown in the proof of Proposition 4.2.2 about this fixed false alarm rate thresholding policy.

Theorem 4.4.3. *Under fixed FAR thresholding policy and for the optimal value of β given from above, the SE has the following phase transition:*

$$\rho_{SE}(\delta; \lambda_{FAR}^*) = \max_z \left(\frac{1 - 2/\delta[(1 + z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - 2[(1 + z^2)\Phi(-z) - z\phi(z)]} \right), \quad (4.25)$$

which is exactly the same as the SE phase transition of minimum risk thresholding policy. Also the optimal value of β is equal to the maximizing z^* in (4.25).

The following corollary is clear from the Theorems 4.4.3 and 4.4.2.

Corollary 4.4.5. *The optimized fixed false alarm rate thresholding policy has the same phase transition as the phase transition of the minimum risk thresholding policy and therefore it is also maximin for the original problem $\sup_{\lambda(\sigma)} \inf_{G \in \mathcal{G}_\gamma} \rho_{SE}(\delta, \lambda(\sigma), G)$*

In other words, the above corollary shows that the phase transition that we obtain from such a simple thresholding policy is the same as the phase transition of the minimum risk thresholding policy. Figure 3.4 shows the optimal values of β as a function of δ . The final remark about this algorithm is the estimation of the σ at each iteration. Although there are many ways to do that, here we explain one simple, yet efficient way which is the median of the absolute value. Since x_o is sparse and at each iteration we observe $x_o + \sigma z$ where $z \sim N(0, I_{N \times N})$ in order to estimate σ we can calculate the median of $|x_o + \sigma z|$. The median is robust to outliers and hence one might assume that it is close to the median of $|\sigma z| \approx 0.6745\sigma$. But, as mentioned before there are many other ways to estimate σ .

Fixing the number of detections

Consider another thresholding policy in which the threshold is set according to the following equation,

$$P(|X + \sigma Z| > \lambda_{FD}(\sigma)) = \alpha, \quad (4.26)$$

where as before $X \sim F \in \mathcal{F}_{\epsilon, \gamma}$ is the sparse signal and $Z \sim N(0, 1)$ is the Gaussian noise. In practice this thresholding policy corresponds to fixing the number of elements that pass the threshold. At each step the threshold is set to the magnitude of the ℓ^{th} largest coefficient (in the absolute value). A good aspect of this thresholding policy is that no estimation of the noise variance is involved in this thresholding policy. Again the parameter α that corresponds to the parameter ℓ is a free parameter that may be optimized by the maximin framework. The following theorem gives us a very simple rule for the maximin tuning of this parameter.

Theorem 4.4.4. *For $\alpha = \delta$ the fixed number of detections thresholding policy achieves the same phase transition as the phase transition of the minimum risk thresholding policy.*

The proof of this theorem is omitted from the dissertation. It is basically similar to the proof of Theorem 4.4.3, and the main difference is that the calculations needed for proving concavity are more complicated. It is also clear that we are not able to improve this phase transition with any thresholding policy according to Theorem 4.4.2 and therefore this thresholding policy can also be considered as maximin for the original problem i.e. $\sup_{\lambda(\sigma)} \inf_{F \in \mathcal{F}_{\epsilon, \gamma}} \rho_{SE}(\delta, \lambda(\sigma), F)$. In practice, the optimal value of α corresponds to the case where $\ell = n$ elements pass the threshold at each iteration. Therefore at each iteration we can easily set the threshold to the magnitude of the n^{th} largest coefficient in absolute value.

One of the main questions that is left unanswered yet is the reason that all these three thresholding policies exhibit the same phase transition. The following lemma shows the main similarity among these three thresholding policies.

Lemma 4.4.5. *Let $\lambda_{MR}^*(\sigma)$, $\lambda_{FAR}^*(\sigma)$ and $\lambda_{FD}^*(\sigma)$ be the maximin optimally tuned value of λ in minimum risk, fixed false alarm rate and fixed number of detections thresholding policies, respectively. Suppose $X \sim F$ and $F = (1 - \epsilon)\delta_0 + \epsilon G$ where G is continuous at 0 and $\epsilon = \rho^*\delta$ where ρ^* is the value of ρ at the phase transition. Then,*

$$\begin{aligned} \lim_{\sigma \rightarrow 0} \mathbb{P}(\eta(X + \sigma Z; \lambda_{MR}^*(\sigma)) > 0) &= \\ \lim_{\sigma \rightarrow 0} \mathbb{P}(\eta(X + \sigma Z; \lambda_{FAR}^*(\sigma)) > 0) &= \\ \lim_{\sigma \rightarrow 0} \mathbb{P}(\eta(X + \sigma Z; \lambda_{FD}^*(\sigma)) > 0) &= \delta. \end{aligned} \quad (4.27)$$

Proof. For the fixed detection thresholding policy it is clear from the definition. Also, in Appendix E.2.4 we showed that $\lim_{\sigma \rightarrow 0} \frac{\lambda_{MR}^*(\sigma)}{\lambda_{FAR}^*(\sigma)} = 1$ therefore the only thing that we should prove for this lemma is the fixed false alarm rate thresholding policy case. For the simplicity of notation I call $\beta_{FAR}^* = \beta^*$.

$$\begin{aligned} \lim_{\sigma \rightarrow 0} \mathbb{P}(\eta(X + \sigma Z; \lambda_{FAR}^*(\sigma)) > 0) &= 2(1 - \epsilon)\mathbb{P}(\sigma Z > \beta^*\sigma) + \\ \epsilon\mathbb{P}(|X + \sigma Z| > \beta^*\sigma) &= 2(1 - \epsilon)(1 - \Phi(\beta^*)) + \epsilon. \end{aligned} \quad (4.28)$$

By some reparametrization of the phase transition formula it is easy to show,

$$\begin{aligned} \delta &= \frac{\phi(\beta^*)}{\phi(\beta^*) + \beta^*(\Phi(\beta^*) - 1/2)}, \\ \rho^* &= 1 - \frac{\beta^*(1 - \Phi(\beta^*))}{\phi(\beta^*)} \end{aligned}$$

By plugging in these two equations in (4.28) we get

$$2(1 - \epsilon)(1 - \Phi(\beta^*)) + \epsilon = \delta.$$

□

This lemma is suggesting that as $\sigma \rightarrow 0$ all the algorithms are setting the thresholds similarly. Of course for larger values of σ they perform differently. But what

matters for the phase transition is the performance as $\sigma \rightarrow 0$.

4.5 Universality

The SE-derived phase transitions are not sensitive to the detailed distribution of coefficient amplitudes. Empirical results in the simulation section find similar insensitivity of observed phase transitions for MP.

Gaussianity of the measurement matrix A can be relaxed; Simulation section finds that other random matrix ensembles exhibit comparable phase transitions.

In applications, one often uses very large matrices A which are never explicitly represented, but only applied as operators; examples include randomly undersampled partial Fourier transforms. Supporting Information finds that observed phase transitions for MP in the partial Fourier case are comparable to those for random A .

4.6 Simulation results

4.6.1 Data generation

For a given algorithm with a fully specified parameter vector, we conduct one phase transition measurement experiment as follows. We fix a *problem suite*, i.e. a matrix ensemble and a coefficient distribution for generating problem instances (A, x_0) . We also fix a grid of δ values in $[0, 1]$, typically 30 values equispaced between 0.02 and 0.99. Subordinate to this grid, we consider a series of ρ values. Two cases arise frequently:

- *Focused Search design.* 20 values between $\rho_{\text{CG}}(\delta; \chi) - 1/10$ and $\rho_{\text{CG}}(\delta; \chi) + 1/10$, where ρ_{CG} is the theoretically expected phase transition deriving from combinatorial geometry (according to case $\chi \in \{+, \pm, \square\}$).
- *General Search design.* 40 values equispaced between 0 and 1.

We then have a (possibly non-cartesian) grid of δ, ρ values in parameter space $[0, 1]^2$. At each (δ, ρ) combination, we will take M problem instances; in our case $M = 20$.

We also fix a measure of success; see below.

Once we specify the problem size N , the experiment is now fully specified; we set $n = \lceil \delta N \rceil$ and $k = \lceil \rho n \rceil$, and generate M problem instances, and obtain M algorithm outputs \hat{x}_i , and M success indicators S_i , $i = 1, \dots, M$.

A problem instance (y, A, x_0) consists of $n \times N$ matrix A from the given matrix ensemble and a k -sparse vector x_0 from the given coefficient ensemble. Then $y = Ax_0$. The algorithm is called with problem instance (y, A) and it produces a result \hat{x} . We declare success if

$$\frac{\|x_0 - \hat{x}\|_2}{\|x_0\|_2} \leq \mathbf{tol},$$

where \mathbf{tol} is a given parameter; in our case 10^{-4} ; the variable S_i indicates success on the i^{th} Monte Carlo realization. To summarize all M Monte Carlo repetitions, we set $S = \sum_i S_i$.

The result of such an experiment is a dataset with tuples (N, n, k, M, S) ; each tuple giving the results at one combination (ρ, δ) . The meta-information describing the experiment is the specification of the algorithm with all its parameters, the problem suite, and the success measure with its tolerance.

4.6.2 Estimating phase transitions

From such a dataset we find the location of the phase transition as follows. Corresponding to each fixed value of δ in our grid, we have a collection of tuples (N, n, k, M, S) with $n/N = \delta$ and varying k . Pretending that our random number generator makes truly independent random numbers, the result S at one experiment is binomial $\text{Bin}(\pi, M)$, where the success probability $\pi \in [0, 1]$. Extensive prior experiments show that this probability varies from 1 when ρ is well below ρ_{CG} to 0 when ρ is well above ρ_{CG} . In short, the success probability

$$\pi = \pi(\rho|\delta; N).$$

We define the *finite- N phase transition* as the value of ρ at which success probability is 50%:

$$\pi(\rho|\delta; N) = \frac{1}{2} \quad \text{at} \quad \rho = \rho(\delta).$$

This notion is well-known in biometrics where the 50% point of the dose-response is called the LD50. Actually we have the implicit dependence $\rho(\delta) \equiv \rho(\delta|N, \mathbf{tol})$; the tolerance in the success definition has a (usually slight) effect, as well as the problem size N .

To estimate the phase transition from data, we model dependence of success probability on ρ using generalized linear models (GLMs). We take a δ -constant slice of the dataset obtaining triples $(k, M, S(k, n, N))$, and model $S(k, n, N) \sim \text{Bin}(\pi_k; M)$ where the success probabilities obeys a generalized linear model with logistic link

$$\text{logit}(\pi) = a + b\rho$$

where $\rho = k/n$; in biometric language, we are modeling that the dose-response probability, where ρ is the ‘complexity-dose’, follows a logistic curve.

In terms of the fitted parameters \hat{a}, \hat{b} , we have the estimated phase transition

$$\hat{\rho}(\delta) = -\hat{a}/\hat{b},$$

and the estimated transition width is

$$\hat{w}(\delta) = 1/\hat{b}.$$

Note that, actually,

$$\hat{\rho}(\delta) = \hat{\rho}(\delta|N, \mathbf{tol}), \quad \hat{w}(\delta) = \hat{w}(\delta|N, \mathbf{tol}).$$

We may be able to see the phase transition and its width varying with N and with the success tolerance.

Because we make only M measurements in our Monte Carlo experiments, these results are subject to sampling fluctuations. Confidence statements can be made for

$\hat{\rho}$ using standard statistical software.

4.6.3 Tuning of algorithms

The procedure so far gives us, for each fully-specified combination of algorithm parameters Λ and each problem suite \mathcal{S} , a dataset $(\Lambda, \mathcal{S}, \delta, \hat{\rho}(\delta; \Lambda, \mathcal{S}))$. When an algorithm has such parameters, we can define, for each fixed δ , the value of the parameters which gives the highest transition:

$$\hat{\rho}^{opt}(\delta; \mathcal{S}) = \max_{\Lambda} \hat{\rho}(\delta; \Lambda, \mathcal{S});$$

with associated optimal parameters $\Lambda^{opt}(\delta; \mathcal{S})$. When the results of the algorithm depend strongly on problem suite as well, we can also tune to optimize worst-case performance across suites, getting the maximin transition

$$\hat{\rho}^{MM}(\delta) = \max_{\Lambda} \min_{\mathcal{S}} \hat{\rho}(\delta; \Lambda, \mathcal{S}).$$

and corresponding maximin parameters $\Lambda^{MM}(\delta)$. This procedure was followed in Chapter 3 for a wide range of popular algorithms. Figure 4.4 presents the observed maximin transitions.

4.6.4 Empirical phase transition

Figure 4.4 (which is a complete version of Figure 4.3) compares observed phase transitions of several algorithms including AMP. For AMP we used a focused search design, focused around $\rho_{CG}(\delta)$. To reconstruct x , we run $T = 1000$ AMP iterations and report the mean square error at the final iteration. For other algorithms, we used the general search design as described in the previous chapter. For more details about observed phase transitions we refer the reader to Chapter 3.

The calculation of the phase transition curve of AMP takes around 36 hours on a single Pentium 4 processor.

Observed Phase transitions for other coefficient ensembles and matrix ensembles are discussed below in Section 4.6.7 and Section 4.6.8.

4.6.5 Example of the interference heuristic

In Section 4.2, our motivation of the SE formalism used the assumption that the mutual access interference term $\text{MAI}_t = (A^*A - I)(x^t - x_0)$ is marginally nearly Gaussian – i.e. the distribution function of the entries in the MAI vector is approximately Gaussian.

As we mentioned, this heuristic motivates the definition of the MSE map. It is easy to prove that the heuristic is valid at the first iteration; but for the validity of SE, it must continue to be true at every iteration until the algorithm stops. Figure 4.5 presents a typical example. In this example we have considered USE matrix ensemble and Rademacher coefficient ensemble. Also N is set to a small size problem 2000 and $(\delta, \rho) = (0.9, 0.52)$. The algorithm is tracked across 90 iterations. Each panel exhibits a linear trend, indicating approximate Gaussianity. The slope is decreasing with iteration count. The slope is the square root of the MSE, and its decrease indicates that the MSE is evolving towards zero. More interestingly, Figure 4.6 shows the QQplot of the MAI noise for the partial Fourier matrix ensemble. Coefficients here are again from Rademacher ensemble and $(N, \delta, \rho) = (16384, 0.5, 0.35)$.

4.6.6 Testing predictions of state evolution

The last section gave an illustration tracking the actual evolution of the AMP algorithm, it showed that the State Evolution heuristic is qualitatively correct.

We now consider predictions made by SE and their quantitative match with empirical observations. We consider predictions of four observables:

- MSE on zeros and MSE on non-zeros:

$$\begin{aligned} \text{MSEZ} &= \mathbb{E}[\hat{x}(i)^2 | x_0(i) = 0], \\ \text{MSENZ} &= \mathbb{E}[(\hat{x}(i) - x_0(i))^2 | x_0(i) \neq 0] \end{aligned} \quad (4.29)$$

- Missed detection rate and False alarm rate:

$$\begin{aligned} \text{MDR} &= \mathbb{P}[\hat{x}(i) = 0 | x_0(i) \neq 0], \\ \text{FAR} &= \mathbb{P}[\hat{x}(i) \neq 0 | x_0(i) = 0] \end{aligned} \quad (4.30)$$

We illustrate the calculation of MDR. Other quantities are computed similarly. Let $\epsilon = \delta\rho$, and suppose that entries in $x_0(i)$ are either 0, 1, or -1 , with $\mathbb{P}\{x_0(i) = \pm 1\} = \epsilon/2$. Then, with $Z \sim N(0, 1)$,

$$\begin{aligned} \mathbb{P}[\hat{x}(i) = 0 | x_0(i) \neq 0] &= \mathbb{P}[\eta(1 + \frac{\sigma}{\sqrt{\delta}}Z) \neq 0] \\ &= \mathbb{P}[1 + \frac{\sigma}{\sqrt{\delta}}Z \notin (-\lambda\sigma, \lambda\sigma)] \\ &= \mathbb{P}[Z \notin (a, b)] \end{aligned} \quad (4.31)$$

with $a = ((-\lambda - 1/\sigma) \cdot \sqrt{\delta})$, $b = (\lambda - 1/\sigma) \cdot \sqrt{\delta}$.

In short, the calculation merely requires classical properties of the normal distribution. The three other quantities simply require other similar properties of the normal. As discussed before, SE evolution makes an iteration-by-iteration prediction of σ_t ; in order to calculate predictions of MDR, FAR, MSENZ and MSEZ, the parameters ϵ and λ are also needed.

We compared the state evolution predictions with the actual values by a Monte Carlo experiment. We chose these triples (δ, ρ, N) : $(0.3, 0.15, 5000)$, $(0.5, 0.2, 4000)$, $(0.7, 0.36, 3000)$. We again used the standard problem suite (USE matrix and unit amplitude nonzero). At each combination of (δ, ρ, N) , we generated $M = 200$ random problem instances from the standard problem suite, and ran the AMP algorithm for a fixed number of iterations. We computed the observables at each iteration. For example, the empirical missed detection rate is estimated by

$$\text{eMDR}(t) = \frac{\#\{i : x^t(i) = 0 \text{ and } x_0(i) \neq 0\}}{\#\{i : x_0(i) \neq 0\}}.$$

We averaged the observable trajectories across the M Monte Carlo realizations, producing empirical averages.

The results for the three cases are presented in Figures 4.7, 4.8, 4.9. Shown on the display are curves indicating both the theoretical prediction and the empirical averages. In the case of the upper row and the lower left panel, the two curves are so close that one cannot easily tell that two curves are, in fact, being displayed.

4.6.7 Coefficient universality

SE displays invariance of the evolution results with respect to the coefficient distribution of the nonzeros. What happens in practice?

We studied invariance of AMP results as we varied the distributions of the nonzeros in x_0 . We consider the problem $\chi = \pm$ and used the following distributions for the non-zero entries of x_0 :

- Uniform in $[-1, +1]$;
- Radamacher (uniform in $\{+1, -1\}$);
- Gaussian;
- Cauchy.

In this study, $N = 2000$, and we considered $\delta = 0.1, 0.3$. For each value of δ we considered 20 equispaced values of ρ in the interval $[\rho_{\text{CG}}(\delta; \pm) - 1/10, \rho_{\text{CG}}(\delta; \pm) + 1/10]$, running each time $T = 1000$ AMP iterations. Data are presented, respectively, in Figure 4.10.

Each plot displays the fraction of success (S/M) as a function of ρ and a fitted success probability, i.e. in terms of success probabilities, the curves display $\pi(\rho)$. In each case 4 curves and 4 sets of data points are displayed, corresponding to the 4 ensembles. The four datasets are visually quite similar, and it is apparent that indeed a considerable degree of invariance is present.

4.6.8 Matrix universality

In section 4.5 we referred to evidence that our results are not limited to the Gaussian distribution.

We conducted a study of AMP where everything was the same as in Figure 4.3 above, however, the matrix ensemble could change. We considered three such ensembles: USE (columns iid uniformly distributed on the unit sphere), Rademacher (random entries iid ± 1 equiprobable), and Partial Fourier, (randomly select n rows from $N \times N$ fourier matrix.) We only considered the case $\chi = \pm$. Results are shown in Figure 4.11, and compared to the theoretical phase transition for ℓ_1 .

4.6.9 Timing results

In actual applications, AMP runs rapidly.

We first describe a study comparing AMP to the LARS algorithm [49]. For comparison with other first-order methods you may refer to Chapter 7. LARS is appropriate for comparison because, among the iterative algorithms previously proposed, its phase transition is closest to the ℓ_1 transition. So it comes closest to duplicating the AMP sparsity-undersampling tradeoff.

Each algorithm proceeds iteratively and needs a stopping rule. In both cases, we stopped calculations when the relative fidelity measure exceeded 0.999, i.e. when $\|y - Ax^t\|_2 / \|y\|_2 < 0.001$.

In our study, we used the partial Fourier matrix ensemble with unit amplitude for nonzero entries in the signal x_0 . We considered a range of problem sizes (N, n, k) and in each case averaged timing results over $M = 20$ problem instances. Table 4.1 presents timing results.

In all situations studied, AMP is substantially faster than LARS. There are a few very sparse situations – i.e. where k is in the tens or few hundreds – where LARS performs relatively well, losing the race by less than a factor 3. However, as the complexity of the objects increases, so that k is several hundred or even one thousand, LARS is beaten by factors of 10 or even more.

(For very large k , AMP has a decisive advantage. When the matrix A is dense, LARS requires at least $c_1 \cdot k \cdot n \cdot N$ operations, while AMP requires at most $c_2 \cdot n \cdot N$ operations. Here $c_2 = \log((\mathbb{E}X^2)/\sigma_T^2)/b$ is a bound on the number of iterations, and $(\mathbb{E}X^2)/\sigma_T^2$ is the relative improvement in MSE in T iterations. Hence in terms of

Table 4.1: Timing Comparison of AMP and LARS. Average Times in CPU seconds.

N	n	k	AMP	LARS
4096	820	120	0.19	0.7
8192	1640	240	0.34	3.45
16384	3280	480	0.72	19.45
32768	1640	160	2.41	7.28
16384	820	80	1.32	1.51
8192	820	110	0.61	1.91
16384	1640	220	1.1	5.5
32768	3280	440	2.31	23.5
4096	1640	270	0.12	1.22
8192	3280	540	0.22	5.45
16384	6560	1080	0.45	27.3
32768	1640	220	6.95	17.53

flops we have

$$\frac{\text{flops(LARS)}}{\text{flops(AMP)}} \geq \frac{kb(\delta, \rho)}{\log((\mathbb{E}X^2)/\sigma_T^2)}.$$

This logarithmic dependence of the denominator is very weak, and very roughly this ratio scales directly with k .)

We also studied AMP's ability to solve very large problems.

We conducted a series of trials with increasing N in a case where A and A^* can be applied rapidly, without using ordinary matrix storage and matrix operations; specifically, the partial Fourier ensemble. For nonzeros of the signal x_0 , we chose unit amplitude nonzeros.

We considered the fixed choice $(\delta, \rho) = (1/6, 1/8)$ and N ranging from $1K$ ($K = 1024$) to $256K$ in powers of 2. At each signal length N we generated $M = 10$ random problem instances and measured CPU times (on a single Pentium 4 processor) and iteration counts for AMP in each instance. We considered four stopping rules, based on MSE σ^2 , $\sigma^2/2$, $\sigma^2/4$, and $\sigma^2/8$, where $\sigma^2 = 2^{-13}$. We then averaged timing results over the $M = 10$ randomly generated problem instances

Figure 4.12 presents the number of iterations as a function of the problem size and accuracy level. According to the SE formalism, this should be a constant independent

of N at each fixed (δ, ρ) and we see indeed that this is the case for AMP: the number of iterations is close to constant for all large N . Also according to the SE formalism, each additional iteration produces a proportional reduction in formal MSE, and indeed in practice each increment of 5 AMP iterations reduces the actual MSE by about half.

Figure 4.13 presents CPU time as a function of the problem size and accuracy level. Since we are using the partial Fourier ensemble, the cost of applying A and A^* is proportional to $N \log(N)$; this is much less than what we would expect for the cost of applying a general dense matrix. We see that indeed AMP execution time scales very favorably with N in this case – to the eye, the timing seems practically linear with N . The timing results show that each doubling of N produces essentially a doubling of execution time. iteration produces a proportional reduction in formal MSE, and indeed in practice each increment of 5 AMP iterations reduces the MSE by about half. Each doubling of accuracy costs about 30% more computation time.

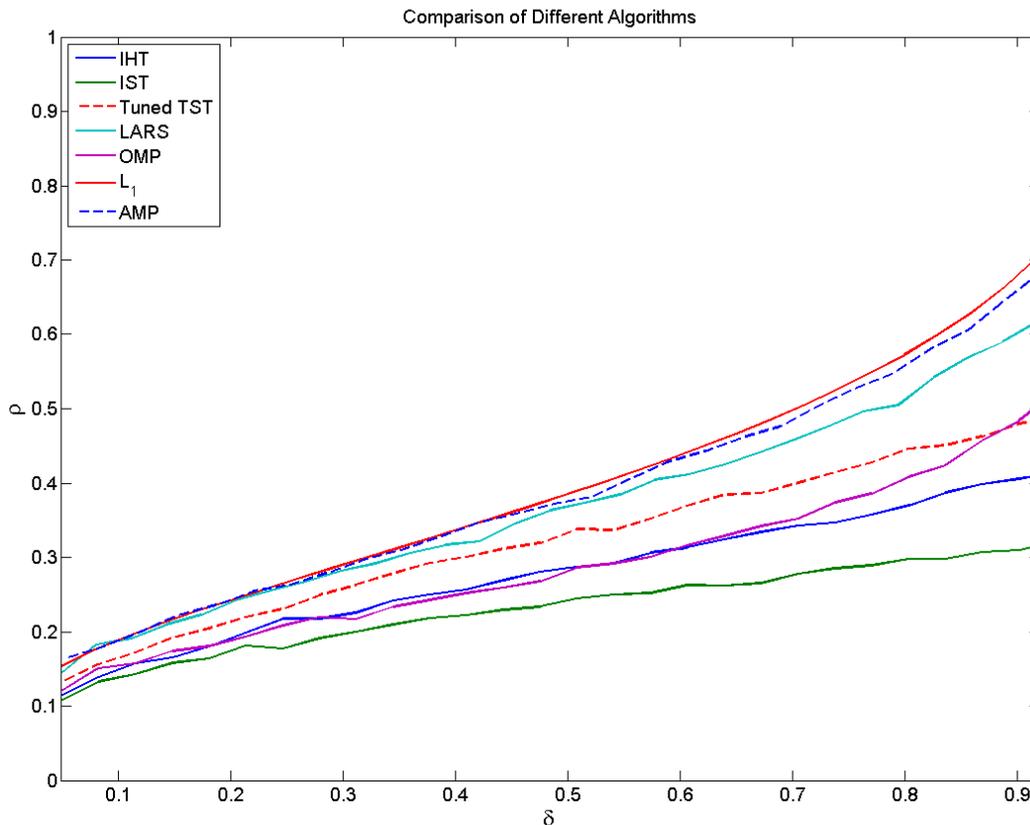


Figure 4.4: Observed Phase Transitions for 6 Algorithms, and ρ_{SE} . AMP: method introduced in main text. IST: Iterative Soft Thresholding. IHT: Iterative Hard Thresholding. TST: a class of two-stage thresholding algorithms including subspace pursuit and CoSaMP. OMP: Orthogonal Matching Pursuit. Note that the ℓ_1 curve coincides with the state evolution transition ρ_{SE} , a theoretical calculation. The other curves show empirical results.

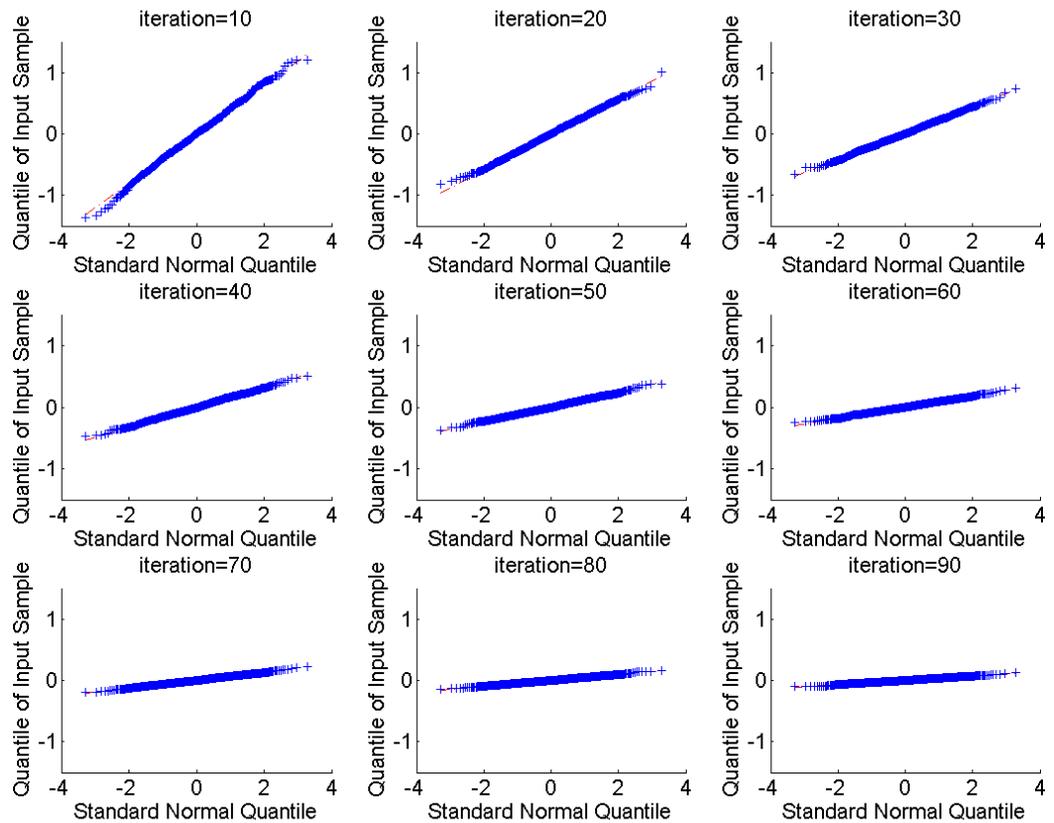


Figure 4.5: QQPlots tracking marginal distribution of mutual access interference (MAI). Panels (a)-(i): iterations 10, 20, \dots , 90. Each panel shows QQplot of MAI values versus normal distribution in blue, and in red (mostly obscured) points along a straight line. Approximate linearity indicates approximate normality. Decreasing slope with increasing iteration number indicates decreasing standard deviation as iterations progress.

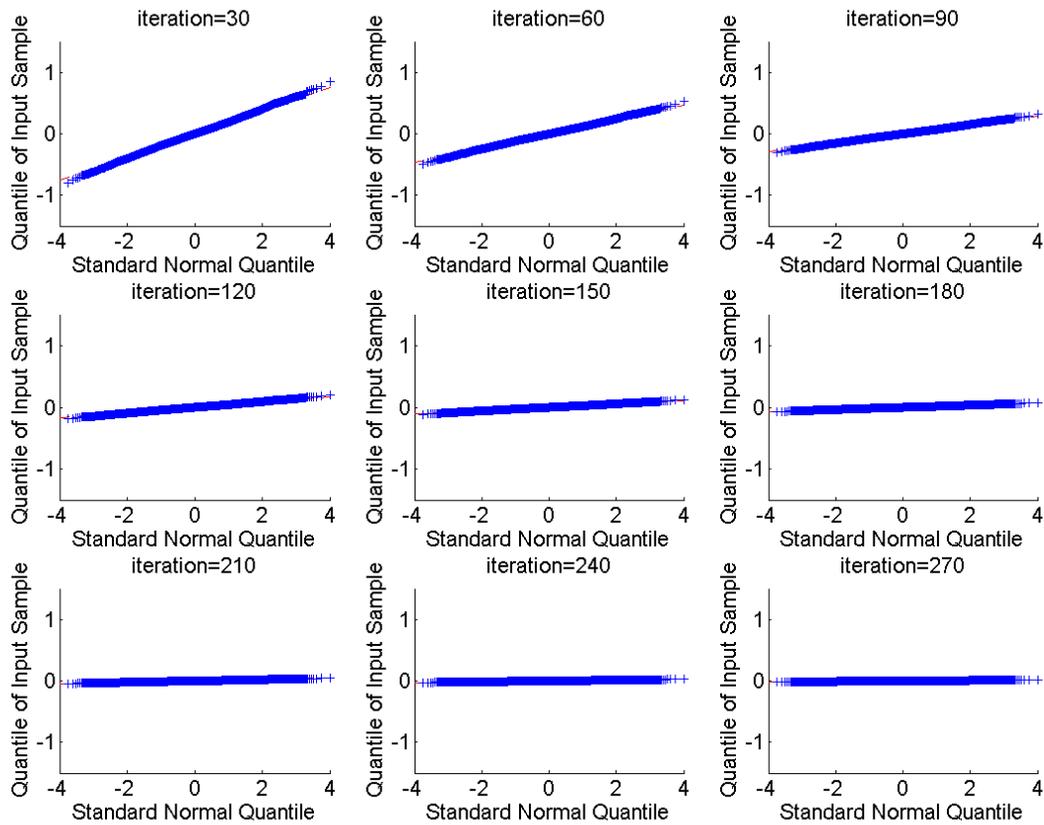


Figure 4.6: QQPlots tracking marginal distribution of mutual access interference (MAI). Matrix Ensemble: partial Fourier. Panels (a)-(i): iterations 30,60,..., 270. For other details, see Figure 4.5.

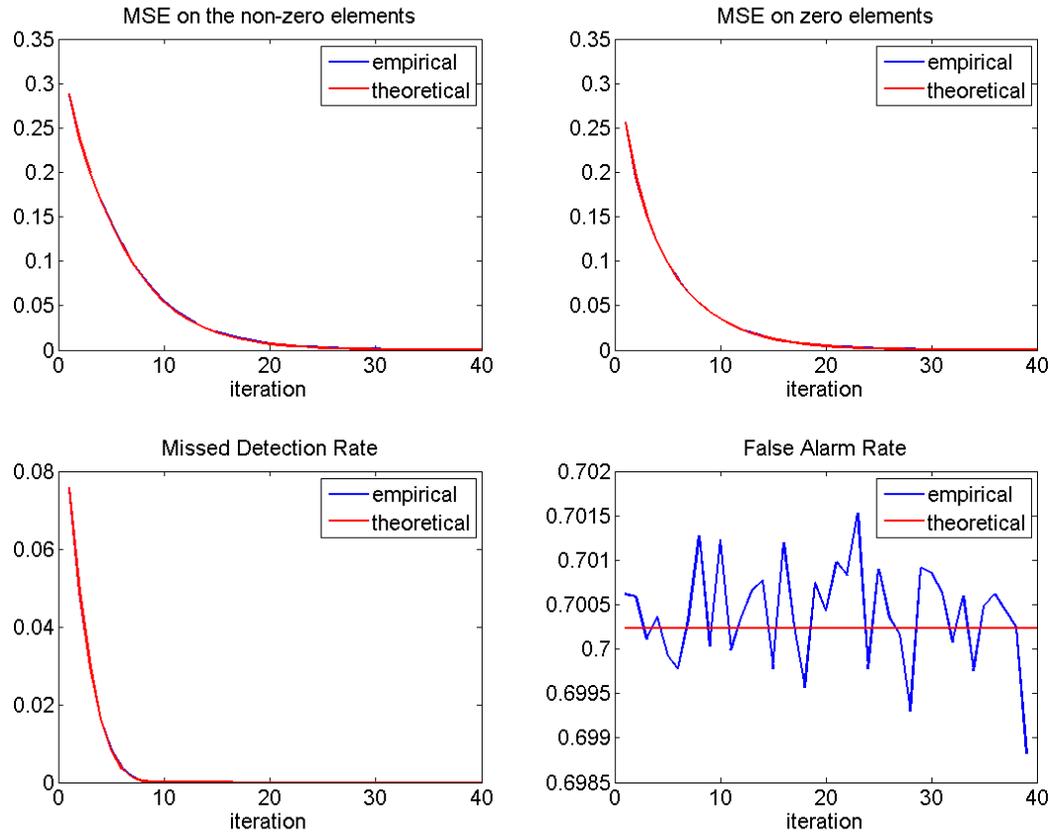


Figure 4.7: Comparison of State Evolution predictions against observations. $\rho = .15$, $\delta = .3$. Panels (a)-(d): MSENZ, MSE, MDR, FAR. Curve in red: theoretical prediction. Curve in blue: mean observable. Each panel shows the evolution of a specific observable as iterations progress. Two curves are present in each panel, however, except for the lower left panel, the blue curve (empirical data) is obscured by the presence of the red curve. The two curves are in close agreement in all panels.

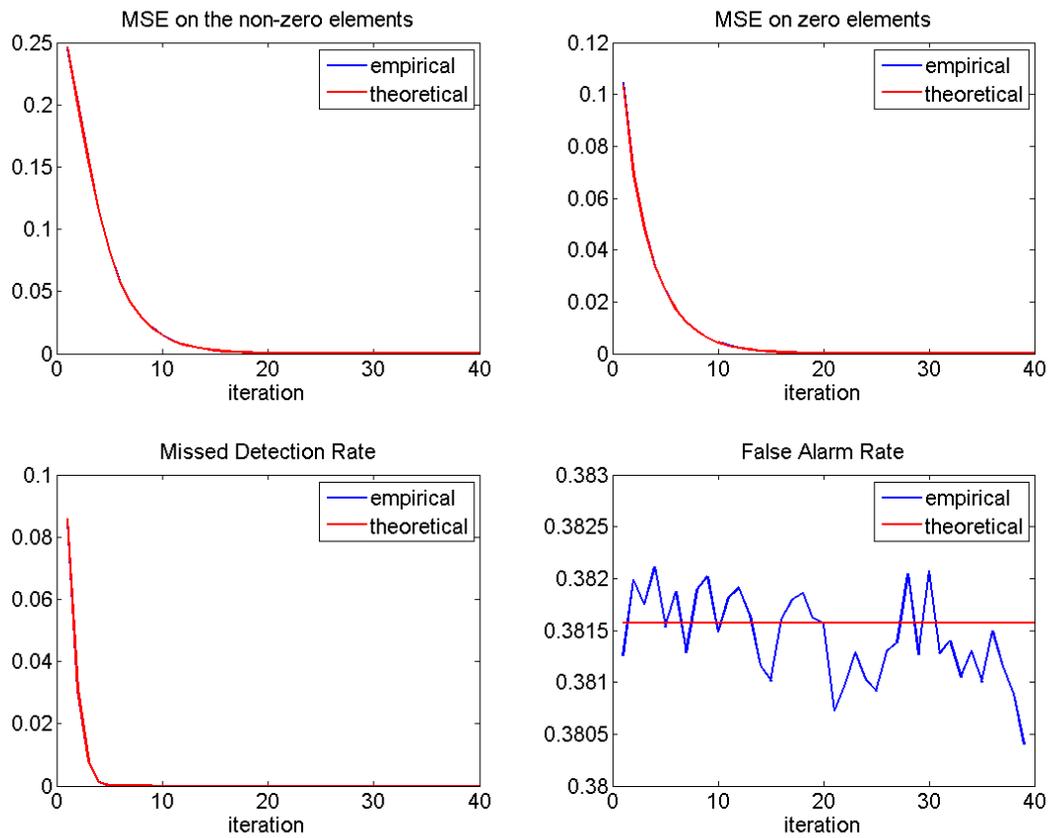


Figure 4.8: Comparison of State Evolution predictions against observations. $\rho = 0.2$, $\delta = 0.5$. For details, see Figure 4.7.

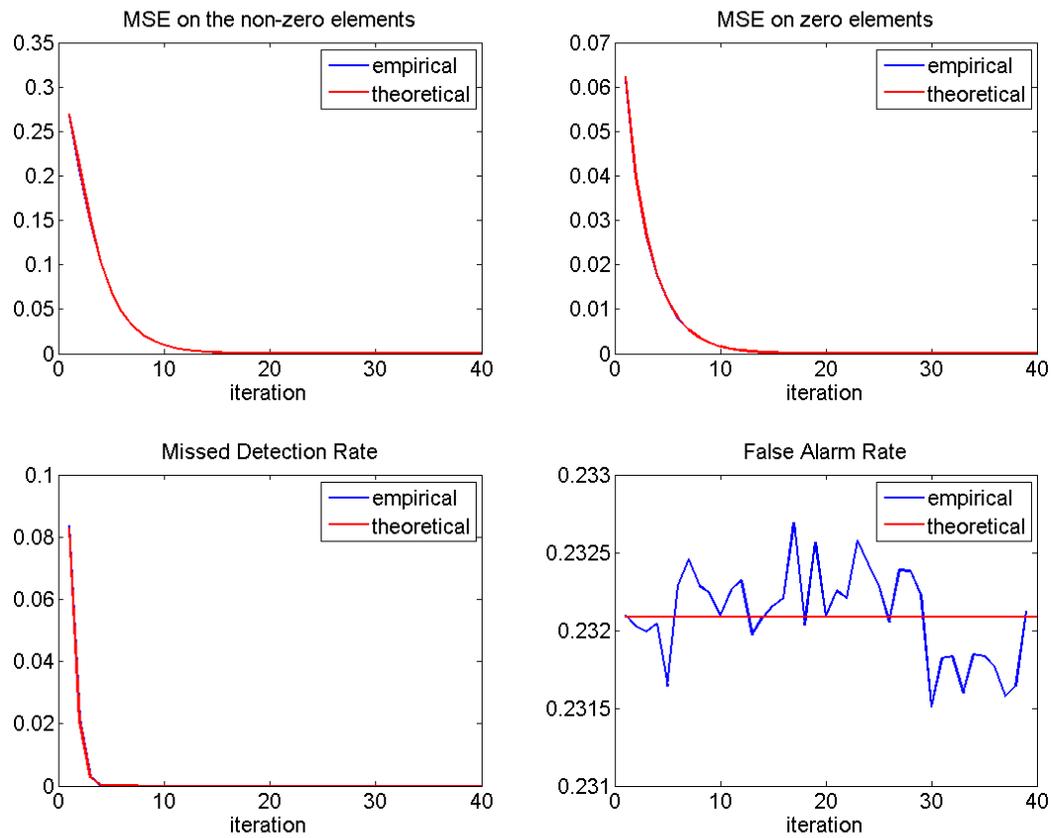


Figure 4.9: Comparison of State Evolution predictions against observations for $\rho = 0.36$, $\delta = 0.7$. For details, see Figure 4.7.

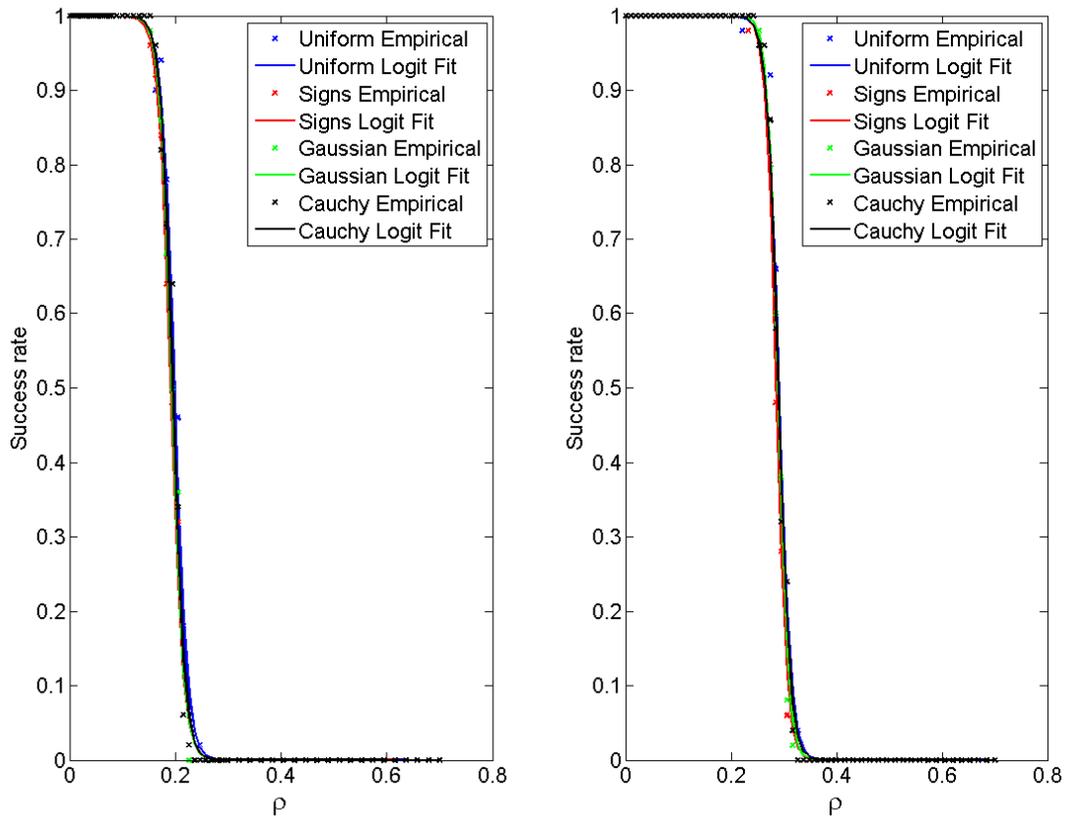


Figure 4.10: Comparison of Failure probabilities for different ensembles. In the left window, $\delta = 0.10$ and in the right window $\delta = 0.3$. Red: unit-amplitude coefficients. Blue: uniform $[-1, 1]$. Green: Gaussian. Black: Cauchy. Points: observed success fractions Curves: Logistic fit.

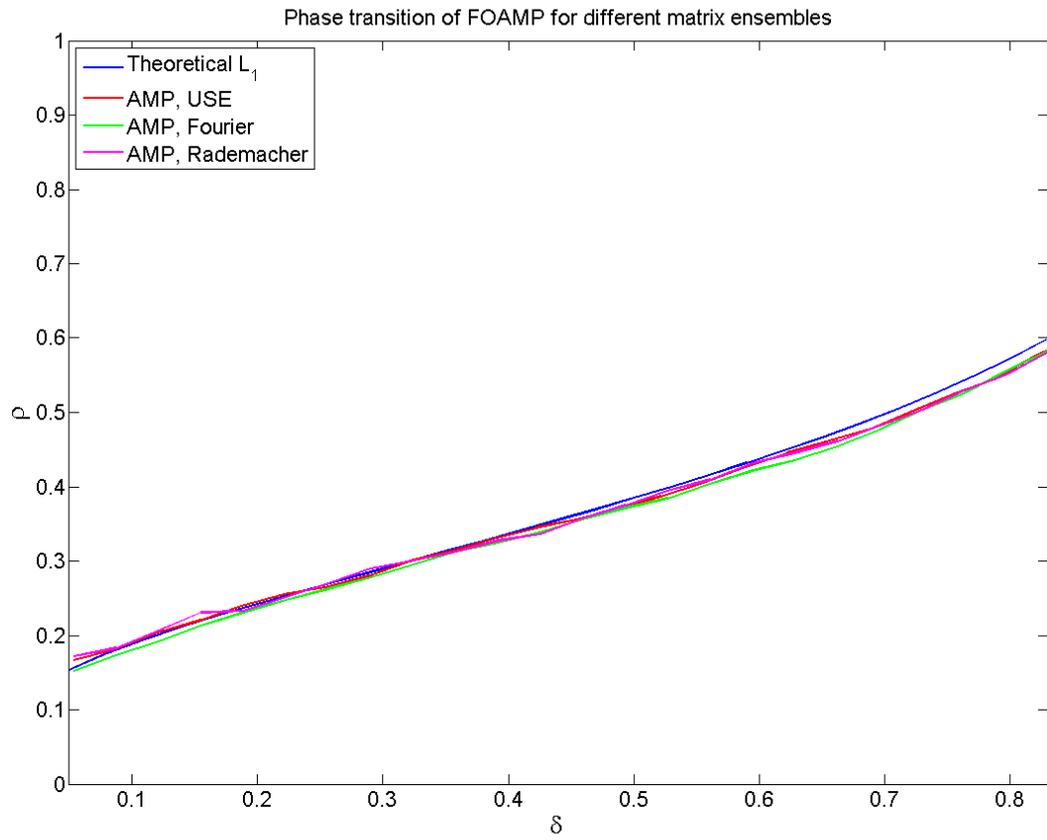


Figure 4.11: Observed Phase Transitions at different matrix ensembles. Case $\chi = \pm$. Red: Uniform Spherical Ensemble (Gaussian with normalized column lengths). Magenta: Rademacher (± 1 equiprobable). Green: partial Fourier. Blue: ρ_{ℓ_1} .

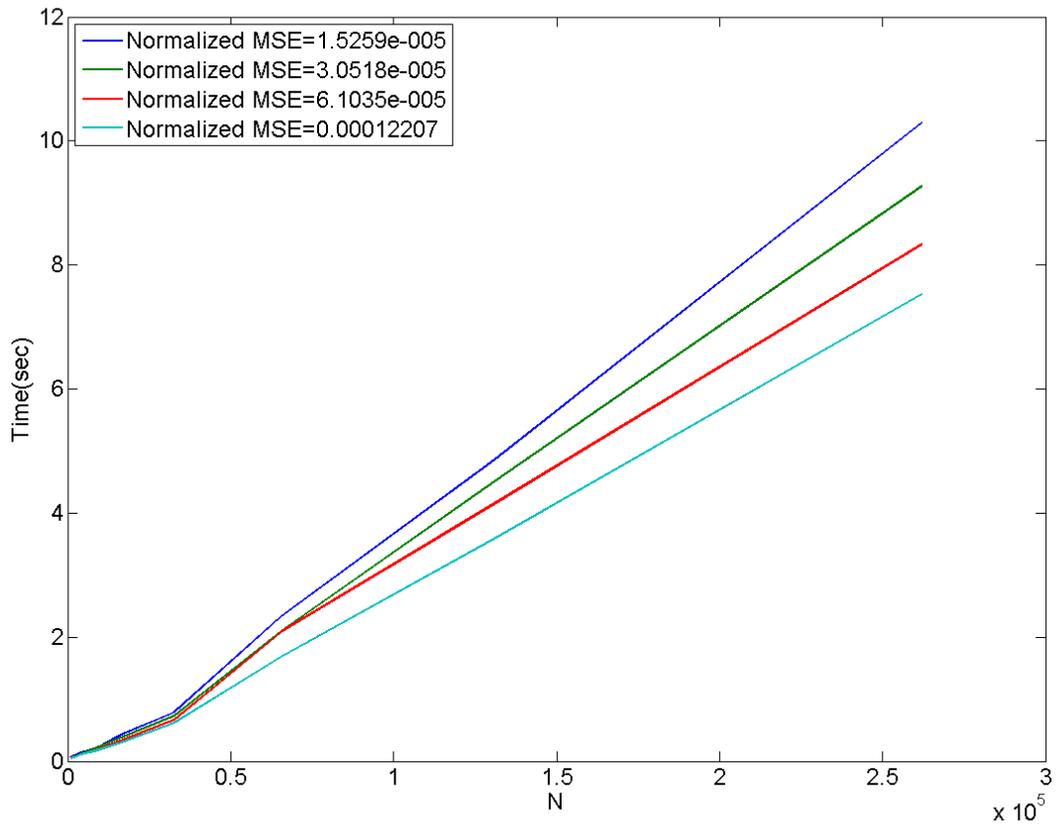


Figure 4.12: Iteration Counts versus Signal Length N . Different curves show results for different stopping rules. Horizontal axis: signal length N . Vertical axis: Number of iterations, T . Blue, Green, Red, Aqua curves depict results when stopping thresholds are set at $2^{-13-\ell}$, with $\ell = 0, 1, 2, 3$. Each doubling of accuracy costs about 5 iterations.

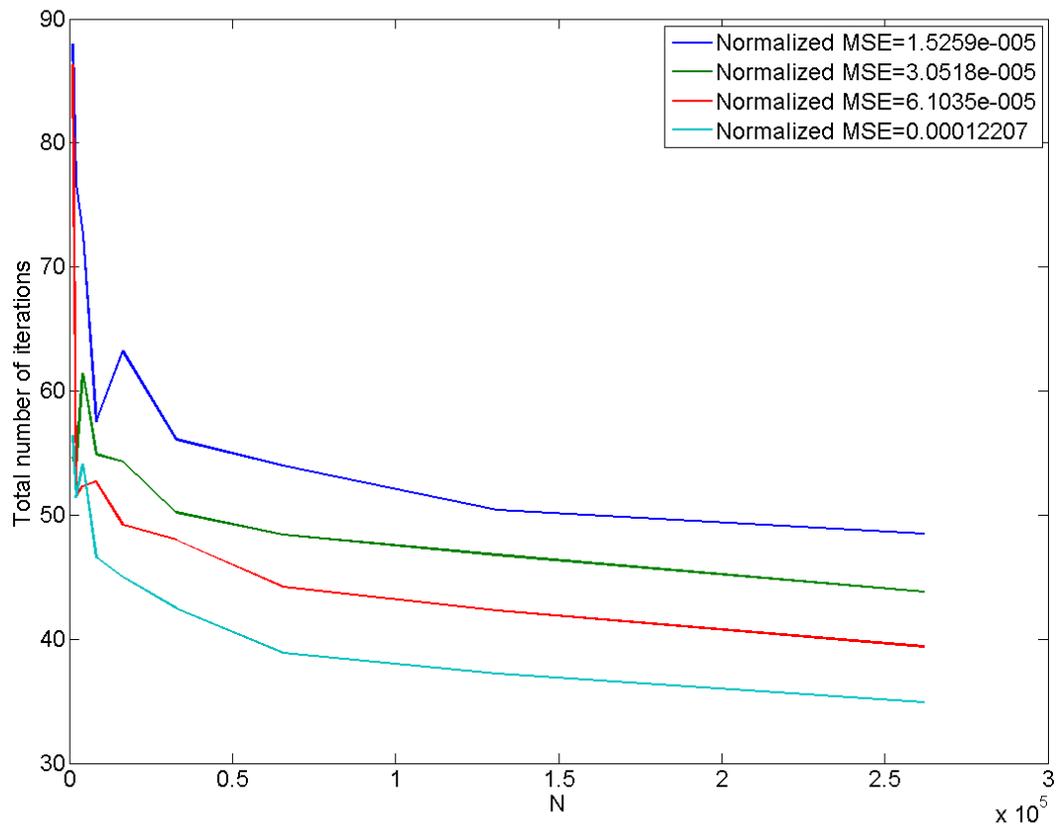


Figure 4.13: CPU Time Scaling with N . Different curves show results for different stopping rules. Horizontal axis: signal length N . Vertical axis: CPU time(seconds). Blue, Green, Red, Aqua curves depict results when stopping thresholds are set at $2^{-13-\ell}$, with $\ell = 0, 1, 2, 3$

Chapter 5

Designing Approximate Message Passing

In the previous chapter we introduced a class of algorithms dubbed AMP, for ‘approximate message passing’, and was inspired by the ideas from graphical models theory, message passing algorithms, and statistical physics. Starting from $x^0 = 0$, the algorithm proceeds according to the following iteration:

$$\begin{aligned}x^{t+1} &= \eta(x^t + A^* z^t; \lambda \hat{\sigma}^t), \\z^t &= y - Ax^t + \langle \eta'(x^{t-1} + A^* z^{t-1}; \lambda \hat{\sigma}^{t-1}) \rangle.\end{aligned}\tag{5.1}$$

η is the thresholding function applied entry-wise. In this chapter, it will be shown that this algorithm is equivalent to the sum product belief propagation algorithm if a suitable joint distribution is considered on the sparse signal. Remarkably, AMP update rules are much simpler than the justifying sum-product rules. Indeed, such a striking simplification emerge in the large system limit. This is one instance of the “blessings of dimensionality” [30]. Apart from justifying the algorithms studied in the previous chapter, the unified derivation provided here allows to develop algorithms for other structures and priors as well. For example, we will consider the following

problem

$$\text{minimize } \lambda \|s\|_1 + \frac{1}{2} \|y - As\|_2^2, \quad (5.2)$$

also known as Basis Pursuit De-Noising (BPDN), or Lasso. We derive an iterative AMP algorithm for this problem that has exactly the same structure as the AMP with different threshold values. Finally, the approach presented here allows to systematically incorporate further information concerning the distribution of the signal s , thus bridging the gap with a Bayesian approach to compressed sensing reconstruction.

5.1 Contribution and organization

In this section we explain the notation that will be used throughout this chapter and will then explain our main contributions. We will also compare the ideas presented here with the related work in the literature.

5.1.1 Notation

Let s_o be a vector in \mathbb{R}^N . We observe $n < N$ linear measurements of this vector through the matrix A , $y = As_o$. As before the goal is to recover s_o from (y, A) . Note that the notation we use for the optimal vector in this chapter is different from the other chapters. This is for avoiding any confusion in the derivations of this chapter. The columns of A are assumed to have unit ℓ_2 -norm. a, b, c, \dots and i, j, k, \dots denote the indices in $[n] \equiv \{1, 2, \dots, n\}$ and $[N] \equiv \{1, 2, \dots, N\}$ respectively. The a, i element of the matrix A will be indicated as A_{ai} . We are following the standard notation in graphical models where a, b, c, \dots represent the factor nodes and i, j, k, \dots are used for the variable nodes [79]. The elements of the vectors y, s, x, s_o are indicated by $y_a, s_i, x_i, s_{o,i}$ respectively. Let $\delta = n/N$ be a measure of indeterminacy of the measurement system. Whenever we refer to the *large system limit* we consider the case where $N \rightarrow \infty$ while δ is fixed. Since in most of the applications of compressed sensing such as the magnetic resonance imaging the problem of interest has millions of variables with tens of thousands of measurements, the large system limits of the

algorithms are of particular interest. In addition large system limit provides very sharp and exact sampling theorems that can then be used for comparing compressed sensing algorithms [76]. It is worth mentioning that in practice, the algorithms we develop in this chapter perform well even in the medium size problems where there are just thousands of variables and hundreds of measurements [38]. In the rest of this section we explain the original problems and the corresponding AMP algorithms.

5.1.2 Basis pursuit problem

Consider the following distribution over the variables s_1, s_2, \dots, s_N

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta|s_i|) \prod_{a=1}^n \delta_{\{y_a=(As)_a\}}, \quad (5.3)$$

where $\delta_{\{y_a=(As)_a\}}$ denotes a Dirac distribution on the hyperplane $y_a = (Ax)_a$. It is clear that as $\beta \rightarrow \infty$, the mass of this distribution concentrates around the solution of basis pursuit. If the minimizer is unique and the marginals of μ are known, the solution of basis pursuit will be immediate. Belief propagation provides a low-complexity heuristic for approximating such marginals. In order to introduce belief propagation, we consider the factor graph $G = (V, F, E)$ with variable nodes $V = [N]$, factor nodes $F = [n]$ and edges $E = [N] \times [n] = \{(i, a) : i \in [N], a \in [n]\}$. Hence G is the complete bipartite graph with N variable nodes and n functional nodes. It is easy to see that the joint distribution (5.3) is structured according to this factor graph.

The state variables of the belief propagation are the messages $\{\nu_{i \rightarrow a}\}_{i \in V, a \in F}$, $\{\hat{\nu}_{a \rightarrow i}\}_{i \in V, a \in F}$ associated with the edges of this graph. In the present case, messages are probability measures over the real line. Throughout this chapter $\nu_{i \rightarrow a}$, $\hat{\nu}_{a \rightarrow i}$ denote densities. The update rules for the densities are

$$\nu_{i \rightarrow a}^{t+1}(s_i) \cong e^{-\beta|s_i|} \prod_{b \neq a} \hat{\nu}_{b \rightarrow i}^t(s_i), \quad (5.4)$$

$$\hat{\nu}_{a \rightarrow i}^t(s_i) \cong \int \prod_{j \neq i} \nu_{j \rightarrow a}^t(s_j) \delta_{\{y_a - (As)_a\}} ds. \quad (5.5)$$

Here and below a superscript denotes the iteration number. Moreover, the symbol \cong denotes identity between probability distributions up to a normalization constant.¹ Unfortunately this message passing algorithm has two problems. First, the messages are density functions over the real line and unless they have certain structure, keeping track of these messages will be very difficult. Second, since the graph is dense the number of messages are $2nN$ and therefore the algorithm is computationally expensive. In Section 5.2 we will prove that in the large system limit and as $\beta \rightarrow \infty$ this complicated message passing algorithm is equivalent to the following simple iterative algorithm.

Starting from $x^0 = 0$ and $\hat{\tau}^0 = 1$ the resulting iterative algorithm proceeds according to

$$\begin{aligned} x^{t+1} &= \eta(A^* z^t + x^t; \hat{\tau}^t), \\ z^t &= y - Ax^t + \frac{1}{\delta} z^{t-1} \langle \eta'(A^* z^{t-1} + x_i^{t-1}; \hat{\tau}^{t-1}) \rangle, \\ \hat{\tau}^t &= \frac{\hat{\tau}^{t-1}}{\delta} \langle \eta'(A^* z^{t-1} + x^t; \hat{\tau}^{t-1}) \rangle, \end{aligned} \quad (5.6)$$

where $\eta(x; b) = \text{sign}(x)(|x| - b)_+$ is the soft thresholding function applied entry-wise. η' is the first derivative of η with respect to the first argument and the notation $\langle \cdot \rangle$ is the averaging operator. Intuitively speaking, the x_i^t in the iterative algorithm corresponds to the mean of the message $\nu_{i \rightarrow a}^t$, z_a^t corresponds to the mean of the message $\hat{\nu}_{a \rightarrow i}^t$ and finally $\hat{\tau}^t$ corresponds to the variance of the message $\nu_{i \rightarrow a}^t$. For more careful definition and analysis of these terms refer to Section 5.2. We will call this algorithm *AMP0*.

5.1.3 BPDN problem

Now consider the following density function over the variables $s = (s_1, \dots, s_N)$.

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta \lambda |s_i|) \prod_{a=1}^n \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} ds. \quad (5.7)$$

¹More precisely, given two non-negative functions $p, q : \Omega \rightarrow \mathbb{R}$ over the same space, we write $p(s) \cong q(s)$ if there exists a positive constant a such that $p(s) = a q(s)$ for every $s \in \Omega$.

Notice that the mode of this distribution coincides with the solution of BPDN and the distribution concentrates on its mode as $\beta \rightarrow \infty$.

In Section 5.3 we will show that each iteration of the sum-product message passing algorithm is equivalent to the following AMP algorithm.

$$\begin{aligned} x^t &= \eta(x^t + A^* z^t; \lambda + \gamma^t), \\ z^{t+1} &= y - Ax^t + \frac{1}{\delta} z^t \langle \eta'(x^{t-1} + A^* z^{t-1}), \rangle \\ \gamma^{t+1} &= \frac{\lambda + \gamma^t}{\delta} \langle \eta'(Az^t + x^t; \gamma^t + \lambda) \rangle. \end{aligned} \quad (5.8)$$

The only difference between this algorithm and AMP0 is in the way the threshold parameter is set. We call this algorithm AMPA where A stands for the automatic threshold selection.

5.1.4 Theoretical prediction

Statistical properties of approximate message passing algorithms allow us to accurately analyze their performance in the asymptotic regime. The state evolution framework introduced in the previous chapter will be briefly reviewed in Section 5.4. Based on this framework we derive the following equations that predict the evolution of AMP0 and AMPA algorithms. Assuming that the empirical distribution of s_o converges weakly to p_s the state evolution equations are

$$\begin{aligned} \tau_{t+1}^2 &= \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_t Z; \lambda + \gamma^t) - X_0]^2, \\ \gamma^{t+1} &= \frac{\gamma^t + \lambda}{\delta} \mathbb{E}[\eta'(X_0 + \tau_t Z; \lambda + \gamma^t)]. \end{aligned}$$

In these two equations (τ^t, γ^t) are called the states of the system at time t . X_0 and Z are two independent random variables with density function p_s and $N(0, 1)$ respectively. σ is the standard deviation of the measurement noise. In the above equations, $\lambda = 0$ corresponds to the AMP0 algorithm.

5.1.5 Extensions

The method we will propose for deriving the above AMP algorithms, enables us to incorporate more complicated priors (if available on the data). To demonstrate this, we consider two more complicated priors in the extension section and develop the corresponding message passing algorithms. First, we will see how one can add a positivity constraint. Second, we will consider an arbitrary product distribution on the variables and will derive a simple iterative algorithm that is equivalent to the sum product belief propagation.

5.1.6 Comparison with other work

First order methods

As mentioned in the introduction finding fast first-order methods for ℓ_1 -minimization is an active area of research and numerous approaches have been proposed [92, 56, 96, 28, 26, 51, 64, 50, 55, 9, 57, 84, 115, 106, 11, 12, 113, 6, 7, 76]. Here we just emphasize on the main differences between the algorithms constructed here with those proposals. For more formal comparison the reader is referred to Chapter 7.

(1) The AMP algorithm is derived from the statistical point of view rather than linear algebraic or convex analysis view point. This makes the accurate analysis of the algorithm on compressed sensing problem possible. The linear algebraic analysis of the convergence rate may provide lower bounds that are far from the reality of compressed sensing problems. For instance, we are able to prove linear convergence of the estimate of AMP to the final solution, while the best result known for linear algebraic methods is strong convergence without any specific bound on the rate [28, 6, 26].

(2) As a result of the statistical analysis all the free parameters can be tuned optimally. Therefore, the algorithms we propose are parameter free. Also the theoretical framework of this algorithm enables us to analyze different continuation strategies [64] which is considered as a difficult problem for other approaches.

Message passing algorithms

The use of message passing algorithms for compressed sensing problems was suggested before, see for instance [93]. However such a proposal faces two major difficulties.

(1) According to the standard prescription, messages used in the sum-product algorithm should be probability measures over the real line \mathbb{R} , cf. (5.10), (5.11). This is impractical from a computational point of view. That's why simpler models such as mixture models are sometimes considered in these cases.

(2) The factor graph on which the sum-product algorithm is run is the complete bipartite graph with N variable nodes, and n function nodes. In other words, unless the underlying matrix is sparse, the graphical model is very dense. This requires to update Nn messages per iteration, and each message update depend on N or n input messages. Again this is very expensive computationally.

(3) The use of belief propagation requires a prior on the vector s_o . However, for most applications, the actual prior is not available.

State evolution and replica calculations

In the context of coding theory, message passing algorithms are analyzed through density evolution [90]. The common justification for density evolution is that the underlying graph is random and sparse, and hence converges locally to a tree in the large system limit. In the case of trees density evolution is exact, hence it is asymptotically exact for sparse random graphs.

State evolution is the analog of density evolution in the case of dense graphs. For definitions and results on state evolution we refer to [38, 41]. The success of state evolution cannot be ascribed to the locally tree-like structure of the graph, and calls for new mathematical ideas.

The fixed points of state evolution describe the output of the corresponding AMP, when the latter is run for a sufficiently large number of iterations (independent of the dimensions n, N). It is well known, within statistical mechanics [79], that the fixed point equations do indeed coincide with the equations obtained through a completely different non-rigorous approach, the *replica method* (in its replica-symmetric form).

This is indeed an instance of a more general equivalence between replica and cavity methods.

During the last year, several papers investigated compressed sensing problems using the replica method [89, 68, 2]. In view of the discussion above, it is not surprising that these results can be recovered from the state evolution formalism put forward in [38]. Let us mention that the latter has several advantages over the replica method: (1) It is more concrete, and its assumptions can be checked quantitatively through simulations; (2) It is intimately related to efficient message passing algorithms; (3) It actually allows to predict the performances of these algorithms.

5.1.7 Organization

In the interest of clarity, we first present our results on the basis pursuit problem in Section 5.2. We will then consider problem (5.2) in Section 5.3. Section 5.4 will be devoted to the asymptotic analysis of the algorithm and finally in Section 5.5 we will be discussing more complicated priors.

5.2 AMP for hard constraints

In the concrete derivation, for the sake of simplicity we assume that $A_{ai} \in \{+1/\sqrt{n}, -1/\sqrt{n}\}$. This is not crucial, and only simplifies some of the calculations. The derivation of AMP proceeds in 4 steps:

1. Construct a joint distribution over (s_1, \dots, s_N) , parameterized by $\beta \in \mathbb{R}_+$, associated with the problem of interest. The distribution is structured according to a graphical model and it is immediate to write down the corresponding sum-product belief propagation algorithm.
2. Show, by central limit theorem argument, that in the large system limit, the sum product messages are well approximated by families with two scalar parameters. Derive the update rules for these parameters.

3. Find the limit $\beta \rightarrow \infty$ (the entire mass of the distribution will concentrate around the mode) and get the appropriate rules for minimization.
4. Approximate the message passing rules for large systems with updates of the form (5.6).

5.2.1 Construction of the graphical model

We consider the following joint probability distribution over the variables s_1, s_2, \dots, s_N

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta|s_i|) \prod_{a=1}^n \mu_{A,y}(ds),$$

where $\mu_{A,y}$ is the Lebesgue measure on the hyperplane $\{s : As = y\}$, and Z is a constant that ensures the normalization $\int \mu(ds) = 1$. In other words, the weights that are assigned to the solutions of the linear system $As = y$, decay exponentially with the ℓ_1 norm of the solutions. This measure can be written more explicitly as

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta|s_i|) \prod_{a=1}^n \delta_{\{y_a=(As)_a\}}. \quad (5.9)$$

Here and below $\delta_{\{y_a=(As)_a\}}$ denotes a Dirac distribution on the hyperplane $y_a = (Ax)_a$. Products of such distributions associated with distinct hyperplanes yield a well defined measure. As we let $\beta \rightarrow \infty$, the mass of the above distribution concentrates around the solution of basis pursuit problem. If the minimizer is unique and the marginals of μ are known, the solution of basis pursuit will be immediate. Belief propagation provides a low-complexity heuristic for approximating marginals.

The update rules for the sum-product message passing algorithm on this graph are

$$\nu_{i \rightarrow a}^{t+1}(s_i) \cong e^{-\beta|s_i|} \prod_{b \neq a} \hat{\nu}_{b \rightarrow i}^t(s_i), \quad (5.10)$$

$$\hat{\nu}_{a \rightarrow i}^t(s_i) \cong \int \prod_{j \neq i} \nu_{j \rightarrow a}^t(s_j) \delta_{\{y_a-(As)_a\}}, \quad (5.11)$$

where superscript denotes the iteration number. In the next section we will try to find the form of the messages in the large system limit.

5.2.2 Large system limit

The main goal in this section is to show that in the large system limit as $N \rightarrow \infty$ the messages have very simple forms. More specifically we show that under certain conditions that will be explained later for n, N large, the messages $\hat{\nu}_{a \rightarrow i}^t(\cdot)$ are approximately Gaussian distributions with variances of order N . On the other hand, the densities of messages $\nu_{i \rightarrow a}^t(\cdot)$ are well approximated by the product of a Gaussian and a Laplace density. We state this fact formally below. Recall that, given two distributions μ_1, μ_2 , their Kolmogorov distance is

$$\|\mu_1 - \mu_2\|_K \equiv \sup_{a \in \mathbb{R}} \left| \int_{-\infty}^a \mu_1(dx) - \int_{-\infty}^a \mu_2(dx) \right|. \quad (5.12)$$

The first Lemma provides an estimate for the messages $\hat{\nu}_{a \rightarrow i}^t$.

Lemma 5.2.1. *Let $x_{j \rightarrow a}^t$ and $(\tau_{j \rightarrow a}^t/\beta)$ be, respectively, the mean and variance of the distribution $\nu_{j \rightarrow a}^t$. Assume further $\int |s_j|^3 d\nu_{j \rightarrow a}^t(s_j) \leq C_t$ uniformly in N, n . Then there exists a constant C'_t such that*

$$\begin{aligned} \|\hat{\nu}_{a \rightarrow i}^t - \hat{\phi}_{a \rightarrow i}^t\|_K &\leq \frac{C'_t}{N^{1/2}(\hat{\tau}_{a \rightarrow i}^t)^{3/2}}, \\ \hat{\phi}_{a \rightarrow i}^t(ds_i) &\equiv \sqrt{\frac{\beta A_{ai}^2}{2\pi \hat{\tau}_{a \rightarrow i}^t}} \exp\left\{\frac{\beta}{2\hat{\tau}_{a \rightarrow i}^t}(A_{ai}s_i - z_{a \rightarrow i}^t)^2\right\} ds_i, \end{aligned} \quad (5.13)$$

where the distribution parameters are given by

$$z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad \hat{\tau}_{a \rightarrow i}^t \equiv \sum_{j \neq i} A_{aj}^2 \tau_{j \rightarrow a}^t. \quad (5.14)$$

Proof. By an easy manipulation, we see that, for any Borel set S

$$\hat{\nu}_{a \rightarrow i}^{t+1}(S) = \mathbb{P}\left\{y_a - \sum_{j \neq i} A_{aj} s_j \in A_{ai} S\right\},$$

where $A_{ai}S = \{A_{ai}x : x \in S\}$. Here probability is over the random vector $(s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_N)$, that is distributed according to the product measure $\nu_{1 \rightarrow a}^t(s_1) \dots \nu_{N \rightarrow a}^t(s_N)$.

Consider the random variable $Z = y_a - \sum_{j \neq i} A_{aj}s_j$. According to the assumptions and the central limit theorem, Z is approximately normal. Clearly

$$\begin{aligned}\mathbb{E}(Z) &= y_a - \sum_{j \neq i} A_{aj}x_{j \rightarrow a}^t, \\ \text{Var}(Z) &= \sum_{j \neq i} A_{aj}^2 \tau_{j \rightarrow a}^t.\end{aligned}$$

The statement follows from Berry-Esseen central limit theorem. \square

Motivated by this lemma, we consider the computation of means and variances of the messages $\nu_{i \rightarrow a}^{t+1}(s_i)$. To state the result, it is convenient to introduce the family of densities

$$f_\beta(s; x, b) \equiv \frac{1}{z_\beta(x, b)} \exp \left\{ -\beta|s| - \frac{\beta}{2b}(s-x)^2 \right\}. \quad (5.15)$$

We also denote as follows its mean and variance (here Z has density $f_\beta(\cdot; x, b)$)

$$\mathbf{F}_\beta(x; b) \equiv \mathbb{E}_{f_\beta(\cdot; x, b)}(Z), \quad \mathbf{G}_\beta(x; b) \equiv \text{Var}_{f_\beta(\cdot; x, b)}(Z). \quad (5.16)$$

Notice that, because of (5.14), $\hat{\tau}_{i \rightarrow a}^t$ is expected to concentrate tightly, and we will therefore assume that it is independent of the edge (i, a) .

Lemma 5.2.2. *Suppose that at iteration t , the messages from factor nodes to the variable nodes are set to $\hat{\nu}_{a \rightarrow i}^t = \hat{\phi}_{a \rightarrow i}^t$, with $\hat{\phi}_{a \rightarrow i}^t$ defined as in (5.13) with parameters $z_{a \rightarrow i}^t$ and $\hat{\tau}_{a \rightarrow i}^t = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu_{i \rightarrow a}^{t+1}(s_i) = \phi_{i \rightarrow a}^{t+1}(s_i) \{1 + O(s_i^2/n)\}, \quad \phi_{i \rightarrow a}^{t+1}(s_i) \equiv f_\beta(s_i; \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t, \hat{\tau}^t). \quad (5.17)$$

In particular, the mean and variances of these messages are given by

$$x_{i \rightarrow a}^{t+1} = F_\beta \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right), \quad \tau_{i \rightarrow a}^t = \beta G_\beta \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right).$$

Proof. (5.17) is simply obtained by pointwise multiplication of the densities $\hat{\phi}_{a \rightarrow i}^t$ in (5.13), according to the general sum-product rule (5.10). More precisely, we obtain

$$\begin{aligned} \nu_{i \rightarrow a}^{t+1}(s_i) &\cong e^{-\beta|s_i|} \prod_{b \neq a} \hat{\nu}_{b \rightarrow i}^t(s_i) = \exp \left\{ -\beta|s_i| - \sum_{b \neq a} \frac{\beta}{2\hat{\tau}^t} (A_{ai}s_i - z_{b \rightarrow i}^t)^2 \right\} \\ &\cong \exp \left\{ -\beta|s_i| - \frac{\beta}{2\hat{\tau}^t} \left(\frac{n-1}{n} s_i^2 - 2s_i \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t \right) \right\}, \end{aligned}$$

which coincides with $\phi_{i \rightarrow a}^{t+1}(s_i)$ up to terms of order s_i^2/n . Finally the formulae for $x_{i \rightarrow a}^{t+1}$ and $\tau_{i \rightarrow a}^t$ follow directly from the definitions of F_β and G_β . \square

Summarizing the above discussion, and approximating $\hat{\tau}_{a \rightarrow i}^t$ with an edge-independent quantity $\hat{\tau}^t$, we reach to the following algorithm.

$$x_{i \rightarrow a}^{t+1} = F_\beta \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.18)$$

$$\hat{\tau}^{t+1} = \frac{\beta}{n} \sum_{i=1}^N G_\beta \left(\sum_b A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right). \quad (5.19)$$

5.2.3 Large β limit

Although we gave a simplified belief propagation formula for a general value of β in the last section, the special case $\beta \rightarrow \infty$ is of particular interest since the mode of the distribution introduced in (5.9) is the same as the Basis pursuit solution. The goal of this section is to derive explicit and simple formulas for the two functions F_β and G_β in the large β limit. Consider the soft threshold function

$$\eta(x; b) = \begin{cases} x - b & \text{if } b < x, \\ 0 & \text{if } -b \leq x \leq b, \\ x + b & \text{if } x < -b. \end{cases} \quad (5.20)$$

It is easy to confirm that,

$$\eta(x; b) = \operatorname{argmin}_{s \in \mathbb{R}} \left\{ |s| + \frac{1}{2b}(s - x)^2 \right\}. \quad (5.21)$$

In the $\beta \rightarrow \infty$ limit, the integral that defines $F_\beta(x; b)$ is dominated by the maximum value of the exponent, that corresponds to $s_* = \eta(x; b)$. Therefore $F_\beta(x; b) \rightarrow \eta(x; b)$ as $\beta \rightarrow \infty$. The variance (and hence the function $F_\beta(x; b)$) can be estimated by approximating the density $f_\beta(s; x, b)$ near s_* . Two cases can occur. If $s_* \neq 0$, then at this point the derivative of the exponent is equal to zero and therefore the density can well be approximate with a Gaussian distribution and $G_\beta(x; b) = \Theta(1/\beta)$. On the other hand if $s_* = 0$, $f_\beta(s; x, b)$ can be approximated by a Laplace distribution, leading to $G_\beta(x; b) = \Theta(1/\beta^2)$. We summarize this discussion in the following lemma:

Lemma 5.2.3. *For bounded x, b , we have*

$$\begin{aligned} \lim_{\beta \rightarrow \infty} F_\beta(x; \beta) &= \eta(x; b), \\ \lim_{\beta \rightarrow \infty} \beta G_\beta(x; \beta) &= b \eta'(x; b). \end{aligned} \quad (5.22)$$

We are therefore led to the following message passing algorithm:

$$x_{i \rightarrow a}^{t+1} = \eta \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.23)$$

$$\hat{\tau}^{t+1} = \frac{\hat{\tau}^t}{N\delta} \sum_{i=1}^N \eta' \left(\sum_b A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right). \quad (5.24)$$

5.2.4 From message passing to AMP

The updates in (5.23), (5.24) are easy to implement but nevertheless the overall algorithm is still computationally expensive since it requires tracking of $2nN$ messages. The goal of this section is to further simplify the message passing update equations. The modification we introduce is expected to become negligible in the large system limit, but reduces the computation cost dramatically.

In order to justify approximation we assume that the messages can be approximated in the following way.

$$\begin{aligned} x_{i \rightarrow a}^t &= x_i^t + \delta x_{i \rightarrow a}^t + O(1/N), \\ z_{a \rightarrow i}^t &= z_a^t + \delta z_{a \rightarrow i}^t + O(1/N), \end{aligned} \quad (5.25)$$

with $\delta x_{i \rightarrow a}^t, \delta z_{a \rightarrow i}^t = O(\frac{1}{\sqrt{N}})$ (here the $O(\cdot)$ errors are uniform in the choice of the edge). We also consider a general message passing algorithms of the form

$$x_{i \rightarrow a}^{t+1} = \eta_t \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t \right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.26)$$

with $\{\eta_t(\cdot)\}_{t \in \mathbb{N}}$ a sequence of differentiable nonlinear functions with bounded derivatives. Notice that the algorithm derived at the end of the previous section, cf. (5.23), (5.24), is indeed of this form, albeit with η_t non-differentiable at 2 points. This does not change the result, as long as the nonlinear functions are Lipschitz continuous. In the interest of simplicity, we shall stick to the differentiable model.

Lemma 5.2.4. *Suppose that the asymptotic behavior (5.25) holds for the message passing algorithm (5.26). Then x_i^t and z_a^t satisfy the following equations*

$$\begin{aligned} x_i^{t+1} &= \eta_t \left(\sum_a A_{ia} z_a^t + x_i^t \right) + o_N(1), \\ z_a^t &= y_a - \sum_j A_{aj} x_j^t + \frac{1}{\delta} z_a^{t-1} \langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1}) \rangle + o_N(1), \end{aligned}$$

where the $o_N(1)$ terms vanish as $N, n \rightarrow \infty$.

Proof. To prove the lemma we substitute (5.25) in (5.26) and write the Taylor expansion of the latter. The update equation for $z_{a \rightarrow i}^t$ yields

$$z_{a \rightarrow i}^t = y_a - \underbrace{\sum_{j \in [N]} A_{aj} x_j^t - \sum_{j \in [N]} A_{aj} \delta x_{j \rightarrow a}^t}_{z_a^t} + \underbrace{A_{ai} x_i^t}_{\delta z_{a \rightarrow i}^t} + O(1/N).$$

For $x_{i \rightarrow a}^{t+1}$ we have

$$x_{i \rightarrow a}^{t+1} = \underbrace{\eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi} \delta z_{b \rightarrow i}^t \right)}_{x_i^t} - A_{ai} z_a^t \underbrace{\eta'_t \left(\sum_{b \in \partial i} A_{bi} z_b^t + \sum_{b \in \partial i} A_{bi} \delta z_{b \rightarrow i}^t \right)}_{\delta x_{i \rightarrow a}^t} + O(1/N).$$

In underbraces we have identified the various contributions. Substituting the expression indicated for $\delta x_{i \rightarrow a}^t$, $\delta z_{a \rightarrow i}^t$ we obtain the recursion for x_i^t and z_a^t . In particular x_i^t is updated according to

$$\begin{aligned} x_i^{t+1} &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi} \delta z_{b \rightarrow i}^t \right) + o(1) \\ &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi}^2 x_i^t \right) + o(1) \\ &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + x_i^t \right) + o(1). \end{aligned}$$

For z_a^t we get

$$\begin{aligned} z_a^t &= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \sum_{j \in [N]} A_{aj}^2 z_a^{t-1} \eta'_{t-1} \left(\sum_{b \in [n]} A_{bj} z_b^{t-1} + \sum_{b \in [n]} A_{aj} \delta z_{a \rightarrow j}^{t-1} \right) + o(1) \\ &= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \frac{1}{n} z_a^{t-1} \sum_{j \in [N]} \eta' \left(\sum_{b \in [n]} A_{bi} z_b^{t-1} + x_i^{t-1} \right) + o(1) \\ &= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \frac{1}{\delta} z_a^{t-1} \langle \eta_{t-1} \left(\sum_{b \in [n]} A_{bi} z_b^{t-1} + x_i^{t-1} \right) \rangle + o(1). \end{aligned}$$

□

This theorem naturally suggest a simplified form of the Iterations (5.23), (5.24). The resulting algorithm can be written in the vector notation as

$$\begin{aligned} x^{t+1} &= \eta(A^* z^t + x^t; \hat{\tau}^t), \\ z^t &= y - Ax^t + \frac{1}{\delta} z^{t-1} \langle \eta'(A^* z^{t-1} + x_i^{t-1}; \hat{\tau}^{t-1}) \rangle, \end{aligned} \quad (5.27)$$

where $\langle \cdot \rangle$ denotes the average entry of a vector.

The recursion for $\hat{\tau}$ is also as follows.

$$\hat{\tau}^t = \frac{\hat{\tau}^{t-1}}{\delta} \langle \eta'(A^* z^{t-1} + x^t; \hat{\tau}^{t-1}) \rangle. \quad (5.28)$$

5.3 AMP for soft constraints

Another popular reconstruction procedure in compressed sensing is the following optimization problem

$$\text{minimize } \lambda \|s\|_1 + \frac{1}{2} \|y - As\|_2^2. \quad (5.29)$$

In this section we describe another approximate message passing algorithm for solving this optimization problem. We will follow closely the four-step procedure already outlined in the previous section. The algorithm that is derived is very similar to the AMP algorithm introduced in the previous section. The only difference is in the update rule for the threshold level.

5.3.1 Construction of the graphical model

As before we define a joint density distribution on the variables $s = (s_1, \dots, s_N)$

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta \lambda |s_i|) \prod_{a=1}^n \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} ds. \quad (5.30)$$

Notice that –as in the previous case– the mode of this distribution coincides with the solution of the relevant problem (5.29). The distribution concentrates on its mode as $\beta \rightarrow \infty$. The sum-product algorithm is

$$\nu_{i \rightarrow a}^{t+1}(s_i) \cong \exp(-\beta \lambda |s_i|) \prod_{b \neq a} \nu_{b \rightarrow i}^t(s_i), \quad (5.31)$$

$$\hat{\nu}_{a \rightarrow i}^t(s_i) \cong \int \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} \prod_{j \neq i} d\nu_{j \rightarrow a}^t(s_j). \quad (5.32)$$

5.3.2 Large system limit

The normal approximation lemma is similar in form to the one given before, with two important differences: (i) The variance of the resulting messages is larger (because the constraint $y_a = (As)_a$ is only enforced softly); (ii) We can approximate the density of $\hat{\nu}_{a \rightarrow i}^t$ with a Gaussian density (not just the corresponding distribution function) which is in fact stronger than the previous result.

Lemma 5.3.1. *Let $x_{j \rightarrow a}^t$ and $(\tau_{j \rightarrow a}^t/\beta)$ be, respectively, the mean and variance of the distribution $\nu_{j \rightarrow a}^t$, for the sum-product algorithm (5.31), (5.32). Assume further $\int |s_j|^3 d\nu_{j \rightarrow a}^t(s_j) \leq C_t$ uniformly in N, n . Then there exists a constant C'_t such that*

$$\sup_{s_i \in \mathbb{R}} |\hat{\nu}_{a \rightarrow i}^t(s_i) - \hat{\phi}_{a \rightarrow i}^t(s_i)| \leq \frac{C'_t}{N(\hat{\tau}_{a \rightarrow i}^t)^{3/2}},$$

$$\hat{\phi}_{a \rightarrow i}^t(s_i) \equiv \sqrt{\frac{\beta A_{ai}^2}{2\pi(1 + \hat{\tau}_{a \rightarrow i}^t)}} \exp\left\{-\frac{\beta}{2(1 + \hat{\tau}_{a \rightarrow i}^t)}(A_{ai}s_i - z_{a \rightarrow i}^t)^2\right\} \quad (5.33)$$

where the distribution parameters are given by

$$z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad \hat{\tau}_{a \rightarrow i}^t \equiv \sum_{j \neq i} A_{aj}^2 \tau_{j \rightarrow a}^t. \quad (5.34)$$

Proof. We have

$$\begin{aligned} \hat{\nu}_{a \rightarrow i}^t(s_i) &\cong \mathbb{E} \exp\left(-\frac{\beta}{2}(y_a - A_{ai}s_i - \sum_{j \neq i} A_{aj}s_j)^2\right) \\ &\cong \mathbb{E} \exp\left(-\frac{\beta}{2}(A_{ai}s_i - Z)^2\right) \\ &\cong \mathbb{E} h_{s_i}(Z), \end{aligned}$$

Here expectation is over $s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_N$ independent and distributed according to $\nu_{1 \rightarrow a}^t, \dots, \nu_{N \rightarrow a}^t$. Further, we defined $Z = y_a - \sum_{j \neq i} A_{aj}s_j$ and $h_{s_i}(z) \equiv \exp(-(\beta/2)(A_{ai}s_i - z)^2)$.

It is not hard to compute the mean and the variance of Z

$$\begin{aligned}\mathbb{E}(Z) &= y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t = z_{a \rightarrow i}^t, \\ \text{Var}(Z) &= \frac{1}{\beta} \sum_{j \neq i} A_{aj}^2 \tau_{j \rightarrow a}^t = \frac{1}{\beta} \hat{\tau}_{a \rightarrow i}^t.\end{aligned}$$

Let W be a normal random variable with the same mean and variance as Z . By a different form of Berry-Esseen central limit theorem mentioned in Appendix B,

$$|\mathbb{E}h_{s_i}(Z) - \mathbb{E}h_{s_i}(W)| \leq \|h'\|_\infty \frac{C_t''}{N^{1/2}(\hat{\tau}_{a \rightarrow i}^t)^{3/2}} \leq \frac{C_t'''}{N^{1/2}(\hat{\tau}_{a \rightarrow i}^t)^{3/2}},$$

where $\|h'\|_\infty = \sup_t |h'(t)|$ is the infinity norm of h' (which is bounded by $\sqrt{\beta}$). We therefore get

$$\begin{aligned}\sup_{s_i \in \mathbb{R}} \left| \hat{\nu}_{a \rightarrow i}^t(s_i) - \mathbb{E}h_{s_i}(W) \right| &\leq \left| \frac{\mathbb{E}h_{s_i}(Z)}{\int \mathbb{E}h_{s_i}(Z) ds_i} - \frac{\mathbb{E}h_{s_i}(W)}{\int \mathbb{E}h_{s_i}(W) ds_i} \right| \\ &\leq \left| \frac{\mathbb{E}h_{s_i}(Z) - \mathbb{E}h_{s_i}(W)}{\int \mathbb{E}h_{s_i}(Z) ds_i} \right| + \left| \frac{\int \mathbb{E}h_{s_i}(W) ds_i - \int \mathbb{E}h_{s_i}(Z) ds_i}{\int \mathbb{E}h_{s_i}(W) ds_i \int \mathbb{E}h_{s_i}(Z) ds_i} \mathbb{E}h_{s_i}(Z) \right| \\ &\leq \frac{C_t'}{N(\hat{\tau}_{a \rightarrow i}^t)^{3/2}}.\end{aligned}$$

The last inequality is due to the following facts,

$$\begin{aligned}\int \mathbb{E}h_{s_i}(Z) ds_i &= \mathbb{E} \int h_{s_i}(Z) ds_i = \frac{\sqrt{2\pi}}{\sqrt{A_{ai}^2 \beta}}, \\ \int \mathbb{E}h_{s_i}(W) ds_i &= \frac{\sqrt{2\pi}}{\sqrt{A_{ai}^2 \beta}}.\end{aligned}$$

The proof is completed by computing $\mathbb{E}h_{s_i}(W)$. Such computation amounts to a straightforward Gaussian integral, yielding $\mathbb{E}h_{s_i}(W) \cong \hat{\phi}_{a \rightarrow i}^t(s_i)$. \square

The update rule for variable-to-factor node messages, cf. (5.31), is identical to the one used in the case of hard constraints, cf. (5.10), apart from the factor λ in the exponent. Keeping track of this term we obtain the following result.

Lemma 5.3.2. *Suppose that at iteration t , the messages from factor nodes to the variable nodes are set to $\hat{\nu}_{a \rightarrow i}^t = \hat{\phi}_{a \rightarrow i}^t$, with $\hat{\phi}_{a \rightarrow i}^t$ defined as in (5.56). with parameters $z_{a \rightarrow i}^t$ and $\hat{\tau}_{a \rightarrow i}^t = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu_{i \rightarrow a}^{t+1}(s_i) = \phi_{i \rightarrow a}^{t+1}(s_i) \{1 + O(s_i^2/n)\}, \quad \phi_{i \rightarrow a}^{t+1}(s_i) \equiv \lambda f_\beta(\lambda s_i; \lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t, \lambda^2(1 + \hat{\tau}^t)) \quad (5.35)$$

In particular, the mean and variances of these messages are given by

$$x_{i \rightarrow a}^{t+1} = \frac{1}{\lambda} F_\beta\left(\lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)\right), \quad \tau_{i \rightarrow a}^t = \frac{\beta}{\lambda^2} G_\beta\left(\lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)\right),$$

where, f_β, F_β , and G_β are defined in 5.15 and 5.16.

The proof is very similar to the proof of Lemma 5.2.2 and for the sake of brevity we do not mention it here.

As a summary, we get the following simple iterative algorithm which at each iteration equivalent to the corresponding iteration of the message passing algorithm.

$$x_{i \rightarrow a}^{t+1} = \frac{1}{\lambda} F_\beta\left(\lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)\right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.36)$$

$$\hat{\tau}^{t+1} = \frac{\beta}{\lambda^2 n} \sum_{i=1}^N G_\beta\left(\lambda \sum_b A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)\right). \quad (5.37)$$

As before the next step is to derive the algorithm in the limit $\beta \rightarrow \infty$ which is the most interesting regime and is equivalent to basis pursuit denoising problem.

5.3.3 Large β limit

Applying Lemma 5.2.3 to (5.36), (5.37) they reduce –in the large β limit– to

$$x_{i \rightarrow a}^{t+1} = \eta\left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda(1 + \hat{\tau}^t)\right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t,$$

$$\hat{\tau}^{t+1} = \frac{1 + \hat{\tau}^t}{N\delta} \sum_{i=1}^N \eta'\left(\sum_b A_{bi} z_{b \rightarrow i}^t; \lambda(1 + \hat{\tau}^t)\right),$$

where we used the invariance property $\eta(ax; ab) = a\eta(x; b)$ valid for any $a > 0$. If we call $\lambda\hat{\tau}^t = \gamma^t$ the new form of the AMP algorithm is,

$$x_{i \rightarrow a}^{t+1} = \eta\left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda + \gamma^t\right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.38)$$

$$\gamma^{t+1} = \frac{\lambda + \gamma^t}{N\delta} \sum_{i=1}^N \eta'\left(\sum_b A_{bi} z_{b \rightarrow i}^t; \lambda + \gamma^t\right), \quad (5.39)$$

These expression should be compared with (5.38), (5.39) for the basis pursuit algorithm. The only difference is just in the threshold value.

5.3.4 From message passing to AMP

Again, this algorithm can be considerably simplified using the Lemma 5.2.4. In matrix notation we obtain the following equations

$$x^t = \eta(x^t + A^* z^t; \lambda + \gamma^t), \quad (5.40)$$

$$z^{t+1} = y - Ax^t + \frac{1}{\delta} z^t \langle \eta'(x^{t-1} + A^* z^{t-1}), \rangle \quad (5.41)$$

which generalize (5.27) and (5.27). The threshold level is computed iteratively as follows

$$\gamma^{t+1} = \frac{\lambda + \gamma^t}{\delta} \langle \eta'(Az^t + x^t; \gamma^t + \lambda) \rangle. \quad (5.42)$$

5.3.5 Comments

Threshold level. The derivation presented above provides a ‘parameter free’ algorithm. The threshold level $\hat{\tau}^t$ or γ^t is fixed by the recursions (5.28), (5.39). In the basis pursuit problem, one could take the alternative point of view that $\hat{\tau}^t$ is a parameter that can be optimized over. This point of view was adopted in Chapter 4. For the case of Lasso it is again possible to consider the threshold as a free parameter and then tune it such that the fixed point of iteration satisfies the KKT conditions. This approach has been adopted in chapter 6. The analysis and comparison of these thresholding policies are

presented in Section 5.4. We call the AMP algorithm with the thresholding policy introduced in (5.28) and (5.39) AMP0 and AMPA respectively. When we tune the algorithm to get the best phase transition the algorithm is called AMPM where M stands for minimaxity. Finally, when the free parameter is tuned to satisfy the KKT conditions the algorithm is called AMPT.

Mathematical derivation of the AMP. We showed that in a specific limit (large systems, and large β) the sum-product update rules can be considerably simplified to get the update rules (5.27), (5.40). Let us emphasize that our proofs concern just a single step of the iterative procedure. Therefore they do not prove that the (averages and variances) of the sum-product message are precisely tracked by (5.27), (5.40). It could be that the error terms in our approximation, while negligible at each step, conjure up to become large after a finite number of iterations. We do not expect this to happen, but it is nevertheless an open mathematical problem.

5.4 State evolution

In chapter 4 we introduced the state evolution framework for analyzing the performance of the AMP algorithm. This approach has been also rigorously proved recently [4]. Consider the following iterative algorithm

$$\begin{aligned} x^{t+1} &= \eta_t(x^t + A^* z^t), \\ z^t &= y - Ax^t + \frac{1}{\delta} \langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1}) \rangle. \end{aligned} \quad (5.43)$$

where $\eta_t(\cdot)$ is a function that may also depend on the iteration. We recall the following result from [4]. Let $\{A(N)\}$ be a sequence of sensing matrices $A \in \mathbb{R}^{n \times N}$ indexed by N , with iid entries $A_{ij} \sim N(0, 1/n)$, and assume $n/N \rightarrow \delta$. Consider further a sequence of signals $\{x_0(N)\}_{N \geq 0}$, whose empirical distributions converge to a probability measure p_{X_0} on \mathbb{R} with bounded $(2k - 2)^{\text{th}}$ moment, and assume $\mathbb{E}_{\hat{p}}(X_0^{2k-2}) \rightarrow \mathbb{E}_{p_{X_0}}(X_0^{2k-2})$ as $N \rightarrow \infty$ for some $k \geq 2$.

Theorem 5.4.1. *For any pseudo-Lipschitz function $\psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ we have,*

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \psi(x_i^{t+1}, x_{0,i}) = \mathbb{E}[\psi(\eta_t(X_0 + \tau_t Z), X_0)],$$

with $X_0 \sim p_{X_0}$ and $Z \sim N(0, 1)$ independent.

According to the above theorem we can consider the parameter τ_t as the state of the algorithm and track the behavior of this state variable across iterations. If we consider the measurements to be of the form $y = Ax + w$ with $w \sim N(0, \sigma^2 I_n)$, the state evolution equation is given by,

$$\tau_{t+1}^2 = \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta_t(X_0 + \tau_t Z) - X_0]^2, \quad (5.44)$$

where again $X_0 \sim p_{X_0}$ and $Z \sim N(0, 1)$ independent.

Although the state evolution equation has been proved for the case of Gaussian measurement matrix, its validity has been carefully verified through extensive simulations for other random matrices [38]. According to the state evolution we can predict the performance of the AMP algorithm theoretically. For the AMPA algorithm the state evolution can be written as,

$$\begin{aligned} \tau_{t+1}^2 &= \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_t Z; \lambda + \gamma^t) - X_0]^2, \\ \gamma^{t+1} &= \frac{\gamma^t + \lambda}{\delta} \mathbb{E}[\eta'(X_0 + \tau_t Z; \lambda + \gamma^t)]. \end{aligned} \quad (5.45)$$

Figure (5.1) shows the match between the predictions of the state evolution and the Monte Carlo simulation results.

5.4.1 Exactly sparse solution

Suppose there is no measurement noise in the system, i.e. $y = As_o$. Also the elements of s_o are drawn from $(1 - \epsilon)\delta_0(s_{oi}) + \epsilon G(s_{oi})$ where G is a density function on \mathbb{R}^+^2 without a point mass at 0 and define $\rho = \epsilon/\delta$. We are interested in the solution of

²This is just for the simplicity of exposition. The results can be easily extended to other cases.

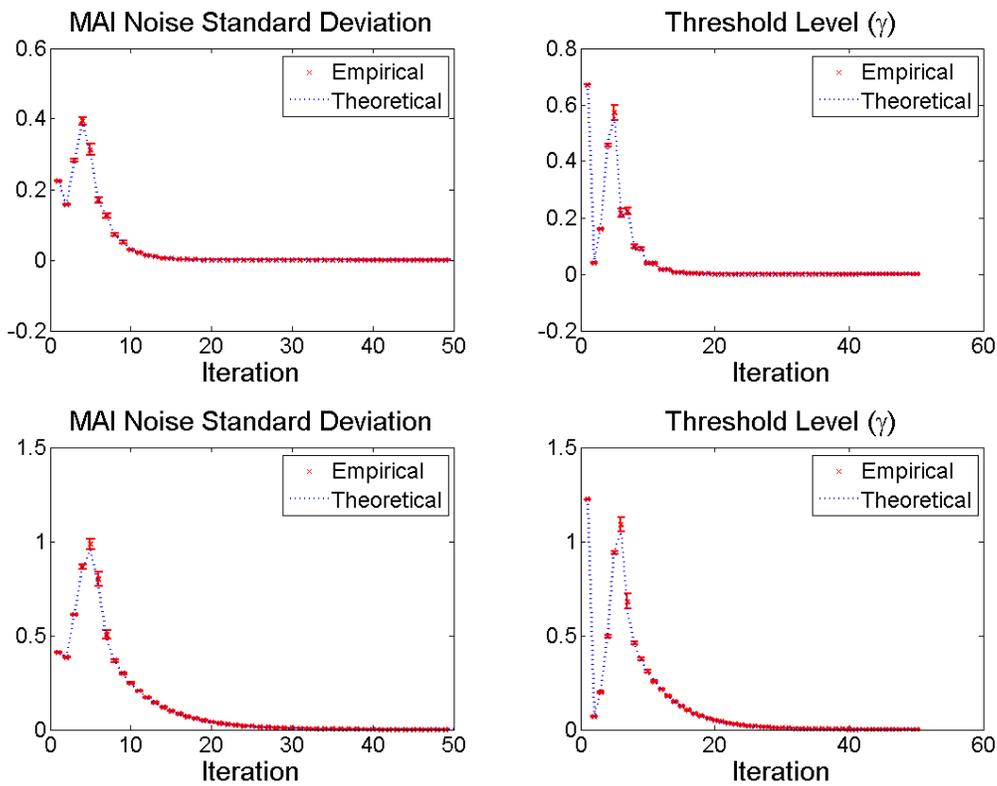


Figure 5.1: Comparison of state evolution predictions against observations. Top row: $N = 4000$, $\delta = .2$ and $\rho = 0.05$. Bottom row: $N = 4000$, $\delta = .3$ and $\rho = 0.17$. Each red point is the average of 50 Monte Carlo samples and the bars show the 95 percent confidence interval.

the basis pursuit problem and therefore we consider the AMP0 algorithm. It is easy to see that the state evolution equation for this case is given by,

$$\begin{aligned}\tau_{t+1}^2 &= \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_t Z; \gamma^t) - X_0]^2, \\ \gamma^{t+1} &= \frac{\gamma^t}{\delta} \mathbb{E}[\eta'(X_0 + \tau_t Z; \gamma^t)].\end{aligned}\tag{5.46}$$

Lemma 5.4.2. *Consider the state evolution 5.46. Suppose that the sparsity level ϵ is small enough such that the algorithm converges to the correct answer, i.e. $(\tau_t, \gamma_t) \rightarrow (0, 0)$. Then*

$$\lim_{t \rightarrow \infty} \frac{\tau_t}{\gamma_t} = c,\tag{5.47}$$

where c is a finite non-zero constant.

Proof. The proof is by contradiction. The goal is to rule out the two cases $c = 0$ and $c = \infty$. Although the proof is a simple application of dominated convergence theorem, for the sake of clarity we mention the proof here.

case I: $c = \infty$. We know that,

$$\begin{aligned}\frac{\gamma^{t+1}}{\gamma^t} &= \frac{1}{\delta} \mathbb{E} \eta'(X + \tau^t Z; \gamma^t) = \frac{1-\epsilon}{\delta} \mathbb{E} \eta'(\tau^t Z; \gamma^t) + \frac{\epsilon}{\delta} \mathbb{E}_{X \sim G} \eta'(X + \tau^t Z; \gamma^t) = \\ &= \frac{1-\epsilon}{\delta} \mathbb{E} \eta'(Z; \gamma^t / \tau^t) + \frac{\epsilon}{\delta} \mathbb{E}_{X \sim G} \eta'(X + \tau^t Z; \gamma^t).\end{aligned}$$

By taking the limit and using the dominated convergence theorem we get,

$$\lim_{t \rightarrow \infty} \frac{\gamma^{t+1}}{\gamma^t} = \frac{1-\epsilon}{\delta} + \frac{\epsilon}{\delta} = \frac{1}{\delta} > 1,$$

and this means that γ^t is not going to zero which is a contradiction.

Case II: $c = 0$.

$$\begin{aligned}\frac{\tau_{t+1}^2}{\gamma_t^2} &= \frac{1}{\delta} \mathbb{E}[\eta(X_0/\gamma_t + Z\tau_t/\gamma_t; 1) - X_0/\gamma_t]^2 = \\ &= \frac{1-\epsilon}{\delta} \mathbb{E}[\eta(Z\tau_t/\gamma_t; 1)]^2 + \frac{\epsilon}{\delta} \mathbb{E}_{X \sim G}[\eta(X_0/\gamma_t + Z\tau_t/\gamma_t; 1) - X_0/\gamma_t]^2.\end{aligned}\tag{5.48}$$

Clearly we can use the dominated convergence theorem again to get the limit and therefore,

$$\lim_{t \rightarrow \infty} \frac{\tau_{t+1}^2}{\gamma_t^2} = \frac{\epsilon}{\delta} > 0, \quad (5.49)$$

which is again a contradiction. \square

Inspired with the above lemma we can introduce a simpler AMP algorithm.

$$\begin{aligned} x^{t+1} &= \eta(x^t + A^* z^t; \theta \sigma^t), \\ z^t &= y - Ax^t + \frac{1}{\delta} z^{t-1} \langle \eta'(x^{t-1} + A^* z^{t-1}; \theta \sigma^{t-1}) \rangle, \end{aligned} \quad (5.50)$$

where σ^t is the standard deviation of the MAI noise at iteration t and θ is a constant number. This is the algorithm that was proposed in chapter 4. Here θ is a parameter that has to be tuned before applying the algorithm. The state evolution for this algorithm is simpler since there is just one state variable.

$$\tau_{t+1}^2 \mapsto \Psi(\tau_t^2) = \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_t Z; \theta \tau_t) - X_0]^2. \quad (5.51)$$

The performance of these two algorithms is very similar as $\tau_t \rightarrow 0$. This is formally stated in the following lemma.

Lemma 5.4.3. *Suppose that $\lim_{\tau \rightarrow 0} \frac{\gamma}{\tau} \rightarrow c$ we then have,*

$$\lim_{\tau \rightarrow 0} \frac{\mathbb{E}[\eta(X_0 + \tau Z; \gamma) - X_0]^2 / \tau^2}{\mathbb{E}[\eta(X_0 + \tau Z; c\tau) - X_0]^2 / \tau^2} = 1.$$

The proof of this lemma is very simple and is omitted. The following result taken from chapter 4 helps us analyze the performance of the new thresholding policy.

Lemma 5.4.4. *The Ψ function is concave and its derivative at $\tau_t = 0$ is independent of G which is the distribution of the non-zero coefficients .*

Let $\rho = \epsilon/\delta$ and define $\rho_{LS}^*(\delta) \equiv \sup_{\theta} \sup\{\rho : \frac{d\Psi}{d\tau^2}|_0 < 1\}$. The optimal value of θ is represented by θ_{MM} which stands for the maximin. The value of θ_{MM} as a function of

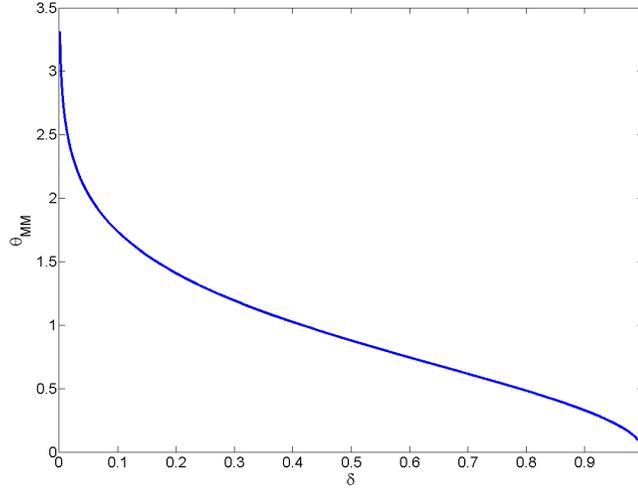


Figure 5.2: The maximin optimal values of θ as proposed in [38].

δ is shown in Figure 5.2. Using the above two lemmas it is easy to prove the following theorem.

Theorem 5.4.5. *If $\rho > \rho_{LS}^*(\delta)$, AMP0 does not converge to the correct answer, i.e. $(\tau_t, \gamma_t) \not\rightarrow (0, 0)$. On the other hand for $\rho < \rho_{LS}^*(\delta)$ the AMPM algorithm converges to the correct answer.*

According to the above theorem from the sparsity measurement point of view AMPM is at least as good as AMP0. The only advantage of AMP0 is that it does not need any tuning. We will show in the next section that for most of the values of δ the phase transitions of AMP0 and AMPA happen at the same place. But for small values of δ , the recursion of AMPA suffer from oscillatory phenomena.

Comparison of AMPA and AMPM

In the previous section we showed that the phase transition of AMP0 algorithm can not surpass the phase transition of AMPM. However the main question is if the state evolution of AMP0 always converges to the correct answer for $\rho < \rho_{LS}^*$. In other words what is the actual phase transition region of the AMPA algorithm? In order to answer

this question precisely, we again use the state evolution equation. We consider 200 equispaced points on the $[0, 1]$ for the values of δ . For each value of δ we also consider 200 equispaced values of ρ . For each pair (δ, ρ) we run the state evolution for 500 iterations and measure the ℓ_2 norm of the estimate after a) 50, (b) 100 (c) 200 and (d) 500 iterations. If $\frac{\|\hat{x}^t - s_o\|_2}{\|s_o\|_2} < .001$, we declare success. With this method we calculate the phase transition of the AMPA algorithm. In this simulation we have chosen the input ensemble from a constant amplitude ensemble which is known to be the least favorable distribution for approximate message passing algorithms [77]. Figure 5.3 compares the phase transition of the AMP0 algorithm derived by this method with the phase transition of AMPM or basis pursuit algorithm. As it is seen in this figure above $\delta > 0.2$ the phase transitions are indistinguishable. However below $\delta = 0.2$ the extra state variable γ causes some instability in the recursive equations that does not exist in AMPM.

5.5 Extensions

So far, we have considered the general compressed sensing problem. However in some applications more information is known about the original signal. In this section we consider two of these scenarios and derive the corresponding approximate message passing algorithms.

5.5.1 Positivity constraint

Suppose that the signal is known to lie in the positive orthant, i.e. $s_{o,i} \geq 0 \quad \forall i$. It has been proved that this extra information may be used properly to improve the phase transition region of the ℓ_1 minimization [45]. This information can be easily incorporated into the message passing algorithm. In this section we just consider the BPDN problem with the above constraint. The BP problem is a very simple modification of this approach and is therefore skipped.

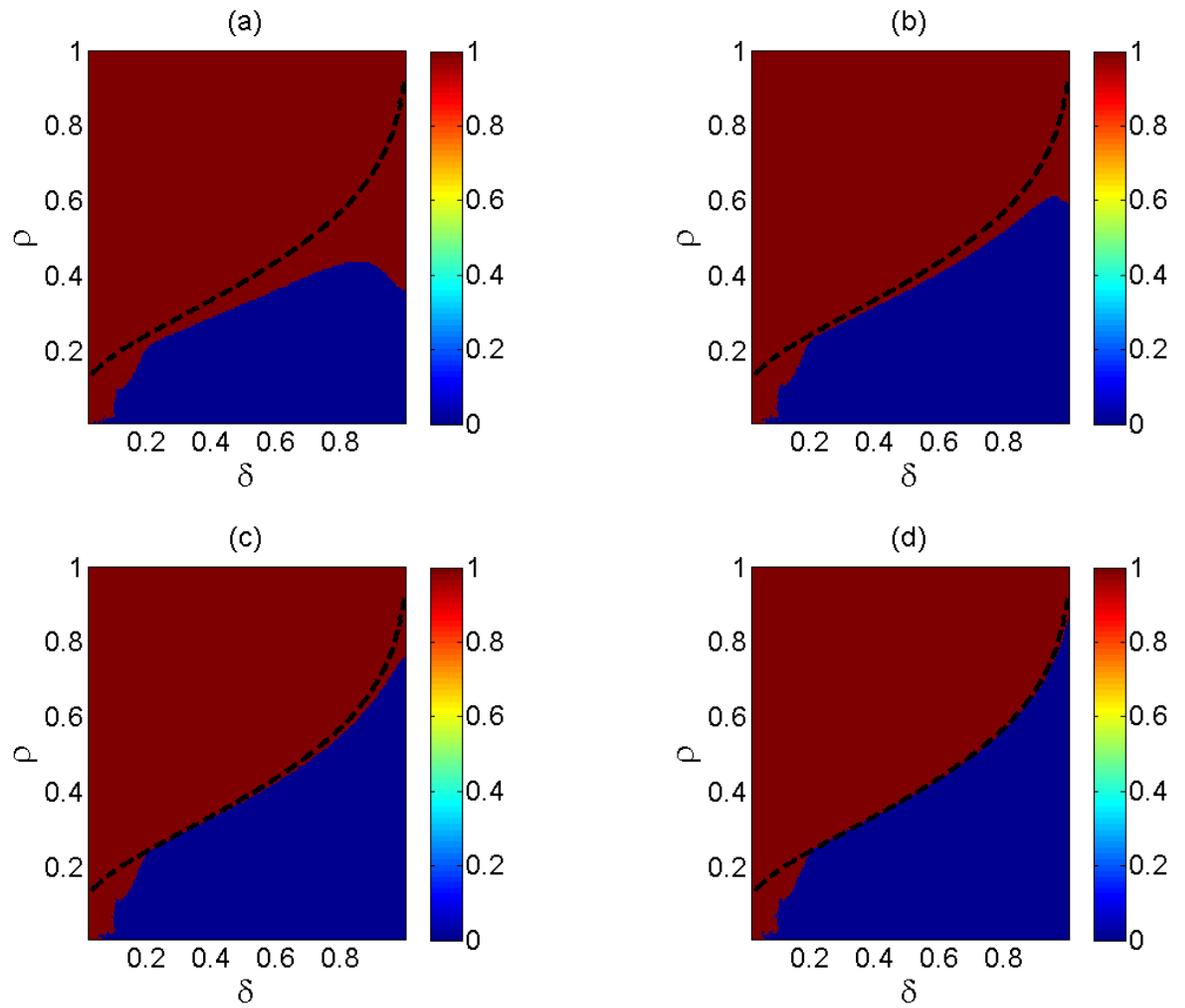


Figure 5.3: Theoretical phase transition of AMPA after (a) 50 (b) 100 (c) 200 and (d) 500 iterations. Dotted line is the phase transition curve of the basis pursuit problem derived in [44] and [38].

Large system limit

Define a joint probability density on the variables $s = (s_1, \dots, s_N)$

$$\mu(ds) = \frac{1}{Z} \prod_{i=1}^N \exp(-\beta \lambda s_i) \mathbb{I}_{\{s_i > 0\}} \prod_{a=1}^n \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} ds. \quad (5.52)$$

As before the messages in the sum-product message passing algorithm are

$$\nu_i^{t+1}(s_i) \cong \exp(-\beta \lambda s_i) \mathbb{I}_{\{s_i > 0\}} \prod_{b \neq a} \nu_{b \rightarrow i}^t(s_i), \quad (5.53)$$

$$\hat{\nu}_{a \rightarrow i}^t(s_i) \cong \int \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} \prod_{j \neq i} d\nu_{j \rightarrow a}^t(s_j). \quad (5.54)$$

Clearly the messages from the functional nodes to the variable nodes have exactly the same form and therefore the following lemma is the immediate result of Theorem 5.3.1.

Lemma 5.5.1. *Let $x_{j \rightarrow a}^t$ and $(\tau_{j \rightarrow a}^t/\beta)$ be, respectively, the mean and variance of the distribution $\nu_{j \rightarrow a}^t$, for the sum-product algorithm (5.31), (5.32). Assume further $\int |s_j|^3 d\nu_{j \rightarrow a}^t(s_j) \leq C_t$ uniformly in N, n . Then there exists a constant C'_t such that*

$$\sup_{s_i \in \mathbb{R}} |\hat{\nu}_{a \rightarrow i}^t(s_i) - \hat{\phi}_{a \rightarrow i}^t(s_i)| \leq \frac{C'_t}{N(\hat{\tau}_{a \rightarrow i}^t)^3}, \quad (5.55)$$

$$\hat{\phi}_{a \rightarrow i}^t(s_i) \equiv \sqrt{\frac{\beta A_{ai}^2}{2\pi(1 + \hat{\tau}_{a \rightarrow i}^t)}} \exp \left\{ -\frac{\beta}{2(1 + \hat{\tau}_{a \rightarrow i}^t)} (A_{ai}s_i - z_{a \rightarrow i}^t)^2 \right\} \quad (5.56)$$

where the distribution parameters are given by

$$z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad \hat{\tau}_{a \rightarrow i}^t \equiv \sum_{j \neq i} A_{aj}^2 \tau_{j \rightarrow a}^t. \quad (5.57)$$

Define

$$f_{\beta}^+(s; x, b) \equiv \frac{1}{z_{\beta}(x, b)} \exp \left\{ -\beta s - \frac{\beta}{2b} (s - x)^2 \right\}, \quad (5.58)$$

and

$$F_{\beta}^{+}(x; b) \equiv \mathbb{E}_{f_{\beta}(\cdot; x, b)}(Z), \quad G_{\beta}^{+}(x; b) \equiv \text{Var}_{f_{\beta}(\cdot; x, b)}(Z). \quad (5.59)$$

It is easy to prove that,

Lemma 5.5.2. *Suppose that at iteration t , the messages from factor nodes to the variable nodes are set to $\hat{\nu}_{a \rightarrow i}^t = \hat{\phi}_{a \rightarrow i}^t$, with $\hat{\phi}_{a \rightarrow i}^t$ defined as in (5.56). with parameters $z_{a \rightarrow i}^t$ and $\hat{\tau}_{a \rightarrow i}^t = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu_{i \rightarrow a}^{t+1}(s_i) = \phi_{i \rightarrow a}^{t+1}(s_i) \{1 + O(s_i^2/n)\}, \quad \phi_{i \rightarrow a}^{t+1}(s_i) \equiv \lambda f_{\beta}^{+}(\lambda s_i; \lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t, \lambda^2(1 + \hat{\tau}^t)) \quad (5.60)$$

In particular, the mean and variances of these messages are given by

$$x_{i \rightarrow a}^{t+1} = \frac{1}{\lambda} F_{\beta}^{+}(\lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)), \quad \tau_{i \rightarrow a}^t = \frac{\beta}{\lambda^2} G_{\beta}^{+}(\lambda \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \lambda^2(1 + \hat{\tau}^t)),$$

where, f_{β}^{+} , F_{β}^{+} and G_{β}^{+} are defined in 5.58 and 5.59.

Large β limit

Consider the following new form of the soft thresholding function.

$$\eta^{+}(x; b) = \begin{cases} x - b & \text{if } b < x, \\ 0 & \text{if } -b \leq x \leq b. \end{cases} \quad (5.61)$$

b in this equation is assumed to be larger than 0. As before when $\beta \rightarrow \infty$, $F_{\beta}^{+}(x; \beta)$ and $G_{\beta}^{+}(x; \beta)$ can be simplified even more.

Lemma 5.5.3. *For bounded x, b , we have*

$$\begin{aligned} \lim_{\beta \rightarrow \infty} F_{\beta}^{+}(x; \beta) &= \eta^{+}(x; b), \\ \lim_{\beta \rightarrow \infty} \beta G_{\beta}^{+}(x; \beta) &= b \eta'^{+}(x; b). \end{aligned}$$

we are therefore led to the following message passing algorithm,

$$x_{i \rightarrow a}^{t+1} = \eta^+ \left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.62)$$

$$\hat{\tau}^{t+1} = \frac{\hat{\tau}^t}{N\delta} \sum_{i=1}^N \eta'^+ \left(\sum_b A_{bi} z_{b \rightarrow i}^t; \hat{\tau}^t \right). \quad (5.63)$$

Finally by similar arguments we can reach to the following approximate message passing algorithm.

$$\begin{aligned} x^{t+1} &= \eta^+(A^* z^t + x^t; \hat{\tau}^t), \\ z^t &= y - Ax^t + \frac{1}{\delta} \langle \eta'(A^* z^{t-1} + x^{t-1}; \hat{\tau}^{t-1}) \rangle, \\ \hat{\tau}^t &= \frac{\hat{\tau}^t}{\delta} \langle \eta'(A^* z^{t-1} + x^{t-1}; \hat{\tau}^{t-1}) \rangle. \end{aligned}$$

5.5.2 AMP for reconstruction with prior information

In many compressed sensing applications it is not realistic to assume that the signal s is random with a known distribution. Nevertheless, it might be possible in specific scenarios to estimate the input distribution. Further, the case of known signal distribution provides a benchmark for other approaches.

Construction of the graphical model

Let $\rho = \rho_1 \times \rho_2 \cdots \times \rho_N$ be a joint probability distribution on the variables s_1, s_2, \dots, s_N . It is then natural to consider the joint distribution

$$\mu(ds) = \frac{1}{Z} \prod_{a=1}^n \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} \prod_{i=1}^N \rho_i(ds_i), \quad (5.64)$$

since μ is the *a posteriori* distribution of s , when $y = As + z$ is observed, with z a noise vector with iid normal entries and independent of s . The sum-product update

rules are

$$\nu_{i \rightarrow a}^{t+1}(\mathrm{d}s_i) \cong \prod_{b \neq a} \hat{\nu}_{b \rightarrow i}^t(s_i) \rho_i(\mathrm{d}s_i), \quad (5.65)$$

$$\nu_{a \rightarrow i}^t(s_i) \cong \int \exp \left\{ -\frac{\beta}{2} (y_a - (As)_a)^2 \right\} \prod_{j \neq i} \nu_{j \rightarrow a}^t(\mathrm{d}s_j). \quad (5.66)$$

Notice that the above update rules are well defined. At each iteration t , the message $\nu_{i \rightarrow a}^{t+1}(\mathrm{d}s_i)$ is a probability measure on \mathbb{R} , and (5.65) gives its density with respect to ρ_i . The message $\nu_{a \rightarrow i}^t(s_i)$ is instead a non-negative measurable function (equivalently, a density) given by (5.66).

It is easy to see that the case studied in the previous section corresponds to choosing the ρ_i 's to be identical exponential distributions.

Large system limit

In view the parallel between the update equations (5.65), (5.66) and (5.31), (5.32) it is easy to realize that Lemma 5.3.1 applies verbatimly to the algorithm described above.

In order to formulate the analogs of Lemma 5.5.2, we introduce the following family of measures over \mathbb{R} :

$$f_i(\mathrm{d}s; x, b) \equiv \frac{1}{z_\beta(x, b)} \exp \left\{ -\frac{\beta}{2b} (s - x)^2 \right\} \rho_i(\mathrm{d}s), \quad (5.67)$$

indexed by $i \in [N]$, $x \in \mathbb{R}$, $b \in \mathbb{R}_+$ (we think β as fixed). We use this notation for its mean and variance (here $Z \sim f_i(\cdot; x, b)$)

$$\mathbf{F}_i(x; b) \equiv \mathbb{E}_{f_i(\cdot; x, b)}(Z), \quad \mathbf{G}_i(x; b) \equiv \mathrm{Var}_{f_i(\cdot; x, b)}(Z). \quad (5.68)$$

These functions have a natural estimation theoretic interpretation. Let X_i be a random variable with distribution ρ_i , and assume that $\tilde{Y}_i = X_i + W_i$ is observed with W_i Gaussian noise with variance b/β . The above functions are –respectively– the

conditional expectation and conditional variance of X_i , given that $\tilde{Y}_i = x$:

$$F_i(x; b) = \mathbb{E}(X_i | \tilde{Y}_i = x), \quad G_i(x; b) = \text{Var}(X_i | \tilde{Y}_i = x). \quad (5.69)$$

With these definitions, it is immediate to prove the following analogous of Lemma 5.5.2.

Lemma 5.5.4. *Suppose that at iteration t , the messages from factor nodes to the variable nodes are set to $\hat{\nu}_{a \rightarrow i}^t = \hat{\phi}_{a \rightarrow i}^t$, with $\hat{\phi}_{a \rightarrow i}^t$ defined as in (5.56). with parameters $z_{a \rightarrow i}^t$ and $\hat{\tau}_{a \rightarrow i}^t = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu_{i \rightarrow a}^{t+1}(s_i) = \phi_{i \rightarrow a}^{t+1}(s_i) \{1 + O(s_i^2/n)\}, \quad \phi_{i \rightarrow a}^{t+1}(s_i) \equiv f_i(s_i; \sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t, (1 + \hat{\tau}^t)). \quad (5.70)$$

In particular, the mean and variances of these messages are given by

$$x_{i \rightarrow a}^{t+1} = F_i\left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; (1 + \hat{\tau}^t)\right), \quad \tau_{i \rightarrow a}^t = \beta G_i\left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; (1 + \hat{\tau}^t)\right).$$

If we let $\hat{\tau}_{i \rightarrow a}^{t+1} = \hat{\tau}^{t+1}$ for all edges (i, a) we get the message passing algorithm

$$x_{i \rightarrow a}^{t+1} = F_i\left(\sum_{b \neq a} A_{bi} z_{b \rightarrow i}^t; (1 + \hat{\tau}^t)\right), \quad z_{a \rightarrow i}^t \equiv y_a - \sum_{j \neq i} A_{aj} x_{j \rightarrow a}^t, \quad (5.71)$$

$$\hat{\tau}^{t+1} = \frac{\beta}{n} \sum_{i=1}^N G_i\left(\lambda \sum_b A_{bi} z_{b \rightarrow i}^t; (1 + \hat{\tau}^t)\right). \quad (5.72)$$

Remarkably, knowledge of the prior distribution is asymptotically equivalent to knowledge of the functions F_i and G_i .

From message passing to AMP

By applying Lemma 5.2.4 we obtain the following algorithm (in matrix notation)

$$x^t = F(x^t + A^* z^t; \lambda + \gamma^t), \quad (5.73)$$

$$z^{t+1} = y - Ax^t + \frac{1}{\delta} z^t \langle F'(x^{t-1} + A^* z^{t-1}) \rangle. \quad (5.74)$$

Here, if $x \in \mathbb{R}^N$, $F(x; b) \in \mathbb{R}^N$ is the vector $F(x; b) = (F_1(x_1; b), F_2(x_2; b), \dots, F_N(x_N; b))$. Analogously $F'(x) = (F'_1(x_1; b), F'_2(x_2; b), \dots, F'_N(x_N; b))$ (derivative being taken with respect to the first argument). Finally, the threshold level is computed iteratively as follows

$$\gamma^{t+1} = \frac{1}{\delta} \langle G(Az^t + x^t; \gamma^t + \lambda) \rangle. \quad (5.75)$$

5.6 Advantages

We presented a step-by-step approach for constructing message passing algorithms. This approach has several advantages:

1. The approach provided here is very general and can be applied to many other settings. The AMP algorithms in general may be slightly more complicated than the AMPA and AMP0 was demonstrated in Section 5.5, but they are much simpler than the actual message passing algorithms on the complete graph.
2. The final approximate message passing algorithm does not have any free parameter. This may come at the cost of more complicated algorithm. The complications may show themselves specially in the analysis as was demonstrated in 5.4.
3. The state evolution framework provides a very simple approach to predict the asymptotic performance of the resulting algorithms.

There are a few open questions that are yet to be answered.

1. The state evolution has been proved to accurately predict the performance of AMP algorithm when the measurement matrix is iid Gaussian. However simulation results show the correctness of state evolution on a wider range of matrix ensembles.
2. Our main concern in the derivation has been the single step performance of the message passing algorithm and not the whole algorithm. Therefore it is

conceivable that the errors accumulate and the algorithm does not perform as well as the actual message passing. The simulation results again confirm that this phenomena does not happen but this has not been addressed theoretically.

Chapter 6

Noise Sensitivity Phase Transition

6.1 Introduction

So far in this thesis we have considered ideal types of compressed sensing problems where the signal is exactly sparse and there is no measurement noise in the system. In this chapter the goal is to consider more realistic assumption i.e. measurement noise is also present in the system. Consider the noisy underdetermined system of linear equations:

$$y = Ax^0 + z^0, \quad (6.1)$$

where the matrix A is $n \times N$, $n < N$, the N -vector x^0 is k -sparse (i.e. it has at most k non-zero entries), and $z^0 \in \mathbb{R}^n$ is a Gaussian white noise $z^0 \sim \mathbf{N}(0, \sigma^2 I)$. Both y and A are known, both x^0 and z^0 are unknown, and we seek an approximation to x^0 .

A very popular approach estimates x^0 via the solution $x^{1,\lambda}$ of the following convex optimization problem

$$(P_{2,\lambda,1}) \quad \text{minimize} \quad \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1. \quad (6.2)$$

Thousands of articles use or study this approach, which has variously been called LASSO, Basis Pursuit, or more prosaically, ℓ_1 -penalized least-squares [103],[24]. There

is a clear need to understand the extent to which $(P_{2,\lambda,1})$ accurately recovers x^0 . Dozens of papers present partial results, setting forth often loose bounds on the behavior of $\hat{x}^{1,\lambda}$. The most well-known analytic approach is the Restricted Isometry Principle (RIP), developed by Candès and Tao [20, 21]. Again in the case where A has iid Gaussian entries, and in the same large-system limit, the RIP implies that, under sufficient sparsity of x^0 , with high probability one has stability bounds of the form $\|\hat{x}^{1,\lambda} - x^0\|_2 \leq C(\delta, \rho)\|z^0\|_2 \log N$. The region where $C(\delta, \rho) < \infty$ was originally an implicitly known, but clearly nonempty region of the (δ, ρ) phase space. Blanchard, Cartis and Tanner [10] recently improved the estimates of C in the case of Gaussian matrices A , by careful large deviations analysis, and by developing an asymmetric RIP, obtaining the largest region where $\hat{x}^{1,\lambda}$ is currently known to be stable. Unfortunately as they show, this region is still relatively small compared to the region $\rho < \rho_{\ell_1}(\delta)$, $0 < \delta < 1$.

It may seem that, in the presence of noise, the precise tradeoff between undersampling and sparsity worsens dramatically, compared to the noiseless case. In fact, the opposite is true. In this chapter, we show that in the presence of Gaussian white noise, the mean-squared error of the optimally tuned ℓ_1 penalized least squares estimator behaves well over quite a large region of the phase plane, in fact, it is finite over the exact same region of the phase plane as the region of $\ell_1 - \ell_0$ equivalence derived in the noiseless case.

Our main results, stated in Section 6.3, give explicit evaluations for the worst-case formal mean square error of $\hat{x}^{1,\lambda}$ under given conditions of noise, sparsity and undersampling. Our results indicate the noise sensitivity of solutions to (6.2), the optimal penalization parameter λ , and the hardest-to-recover sparse vector. As we show, the noise sensitivity exhibits a phase transition in the undersampling-sparsity (δ, ρ) domain along a curve $\rho = \rho_{\text{MSE}}(\delta)$, and this curve is precisely the same as the $\ell_1 - \ell_0$ equivalence curve ρ_{ℓ_1} .

Our results might be compared to work of Xu and Hassibi [114], who considered a different departure from the noiseless case. In their work, the noise z^0 was still vanishing, but the vector x_0 was allowed to be an ℓ_1 -norm bounded perturbation to a k -sparse vector. They considered stable recovery with respect to such small

perturbations and showed that the natural boundary for such stable recovery is again the curve $\rho = \rho_{\text{MSE}}(\delta)$.

6.1.1 Results of our formalism

We define below a so-called formal MSE (fMSE), and evaluate the (minimax, formal) *noise sensitivity*:

$$M^*(\delta, \rho) = \sup_{\sigma > 0} \max_{\nu} \min_{\lambda} \text{fMSE}(\hat{x}^{1,\lambda}, \nu, \sigma^2) / \sigma^2; \quad (6.3)$$

here ν denotes the marginal distribution of x^0 (which has fraction of nonzeros not larger than $\rho\delta$), and λ denotes the tuning parameter of the ℓ_1 -penalized ℓ_2 minimization. Let $M^\pm(\epsilon)$ denote the minimax MSE of scalar thresholding, defined in Section 6.2 below. Let $\rho_{\text{MSE}}(\delta)$ denote the solution of

$$M^\pm(\rho\delta) = \delta. \quad (6.4)$$

Our main *theoretical* result is the formula

$$M^*(\delta, \rho) = \begin{cases} \frac{M^\pm(\delta\rho)}{1 - M^\pm(\delta\rho)/\delta}, & \rho < \rho_{\text{MSE}}(\delta), \\ \infty, & \rho \geq \rho_{\text{MSE}}(\delta). \end{cases} \quad (6.5)$$

Quantity (6.3) is the payoff of a traditional two-person zero sum game, in which the undersampling and sparsity are fixed in advance, the researcher plays against Nature, Nature picks both a noise level and a signal distribution, and the researcher picks a penalization level, in knowledge of Nature's choices. It is traditional in analyzing such games to identify the least-favorable strategy of Nature (who maximizes payout from the researcher), and the optimal strategy for the researcher (who wants to minimize payout). We are able to identify both and give explicit formulas for the so-called saddlepoint strategy, where Nature plays the least-favorable strategy against the researcher and the researcher minimizes the consequent damage. In Proposition 6.3.7 below we give formulas for this pair of strategies. The phase-transition structure

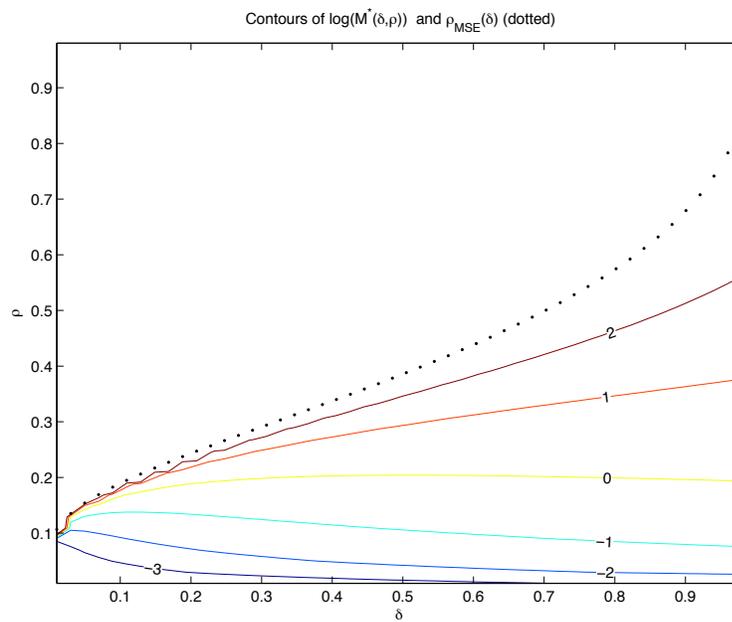


Figure 6.1: Contour lines of the minimax noise sensitivity $M^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted black curve graphs the phase boundary $(\delta, \rho_{\text{MSE}}(\delta))$. Above this curve, $M^*(\delta, \rho) = \infty$. The colored lines present level sets of $M^*(\delta, \rho) = 1/8, 1/4, 1/2, 1, 2, 4$ (from bottom to top).

evident in (6.5) is saying that above the curve ρ_{MSE} , Nature has available unboundedly good strategies, to which the researcher has no effective response. Our approach is based on the formalism we derived for approximate message passing algorithms.

6.2 Minimax MSE of soft thresholding

We briefly recall notions from, e.g., [37, 36] and then generalize them. We wish to recover an N vector $x^0 = (x^0(i) : 1 \leq i \leq N)$ which is observed in Gaussian white noise

$$y(i) = x^0(i) + z^0(i), \quad 1 \leq i \leq N,$$

with $z^0(i) \sim \mathbf{N}(0, \sigma^2)$ independent and identically distributed. This can be regarded as special case of the compressed sensing model (6.1), whereby $n = N$ and $A = I$ is the identity matrix – i.e. there is no underdetermined system of equations. We assume that x^0 is sparse. It makes sense to consider soft thresholding

$$\hat{x}^\tau(i) = \eta(y(i); \tau\sigma), \quad 1 \leq i \leq N,$$

where the soft threshold function (with threshold level θ) is defined by

$$\eta(x; \theta) = \begin{cases} x - \theta & \text{if } \theta < x, \\ 0 & \text{if } -\theta \leq x \leq \theta, \\ x + \theta & \text{if } x \leq -\theta. \end{cases} \quad (6.6)$$

In words, the estimator (6.2) ‘shrinks’ the observations y towards the origin by a multiple τ of the noise level σ .

In place of studying x^0 which are k -sparse, [37, 36] consider random variables X which obey $\mathbb{P}\{X \neq 0\} \leq \epsilon$, where $\epsilon = k/n$. So let \mathcal{F}_ϵ denote the set of probability measures placing all but ϵ of their mass at the origin:

$$\mathcal{F}_\epsilon = \{\nu : \nu \text{ is probability measure with } \nu(\{0\}) \geq 1 - \epsilon\}.$$

We define the soft thresholding mean square error by

$$\text{mse}(\sigma^2; \nu, \tau) \equiv \mathbb{E} \left\{ \left[\eta(X + \sigma \cdot Z; \tau\sigma) - X \right]^2 \right\}. \quad (6.7)$$

Here expectation is with respect to independent random variables $Z \sim \mathbf{N}(0, 1)$ and $X \sim \nu$.

It is important to allow general σ in calculations below. However, note to the scale invariance

$$\text{mse}(\sigma^2; \nu, \tau) = \sigma^2 \text{mse}(1; \nu^{1/\sigma}, \tau), \quad (6.8)$$

where ν^a is the probability distribution obtained by rescaling ν : $\nu^a(S) = \nu(\{x : ax \in S\})$. It follows that all calculations can be made in the $\sigma = 1$ setting and results rescaled to obtain final answers. Below, when we deal with $\sigma = 1$, we will suppress the σ argument, and simply write $\text{mse}(\nu, \tau) \equiv \text{mse}(1; \nu, \tau)$

The *minimax threshold MSE* was defined in [37, 36] by

$$M^\pm(\epsilon) = \inf_{\tau > 0} \sup_{\nu \in \mathcal{F}_\epsilon} \text{mse}(\nu, \tau). \quad (6.9)$$

(The superscript \pm reminds us that, when the estimand X is nonzero, it may take either sign. In Section 6.6.1, the superscript $+$ will be used to cover the case where $X \geq 0$.) We will denote by $\tau^\pm(\epsilon)$ the threshold level achieving the infimum. Figure 6.2 depicts the behavior of M^\pm and τ^\pm as a function of ϵ . $M^\pm(\epsilon)$ was studied in [36] where one can find a considerable amount of information about the behavior of the optimal threshold τ^\pm and the least favorable distribution ν_ϵ^\pm . In particular, the optimal threshold behaves as

$$\tau^\pm(\epsilon) \sim \sqrt{2 \log(\epsilon^{-1})}, \quad \text{as } \epsilon \rightarrow 0,$$

and is explicitly computable at finite ϵ .

A peculiar aspect of this supremum requires us to generalize it somewhat.

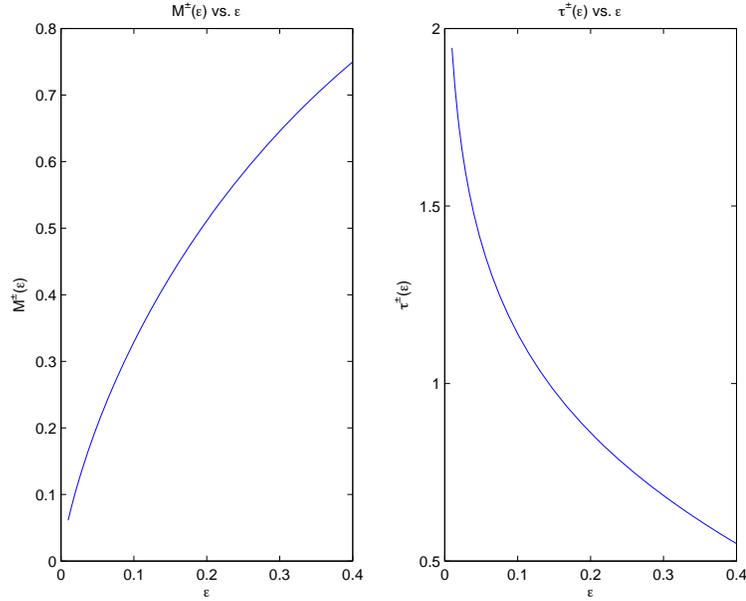


Figure 6.2: Left: $M^\pm(\epsilon)$ as a function of ϵ ; Right: $\tau^\pm(\epsilon)$ as a function of ϵ .

Lemma 6.2.1. *For a given, fixed $\tau > 0$, the worst case MSE obeys*

$$\sup_{\nu \in \mathcal{F}_\epsilon} \text{mse}(\nu, \tau) = \epsilon(1 + \tau^2) + (1 - \epsilon)[2(1 + \tau^2)\Phi(-\tau) - 2\tau\phi(\tau)], \quad (6.10)$$

with $\phi(z) = \exp(-z^2/2)/\sqrt{2\pi}$ the standard normal density and $\Phi(z) = \int_{-\infty}^z \phi(x) dx$ the Gaussian distribution.

The proof of this lemma is based on 4.4.1 and Jensen Inequality. Interestingly, this supremum is “achieved” by a three-point mixture on the *extended* real line $\mathbb{R} \cup \{-\infty, \infty\}$:

$$\nu_\epsilon^* = (1 - \epsilon)\delta_0 + \frac{\epsilon}{2}\delta_\infty + \frac{\epsilon}{2}\delta_{-\infty}.$$

We will need approximations which place no mass at ∞ . We say distribution $\nu_{\epsilon,\alpha}$ is α -*least-favorable* for $\eta(\cdot; \tau)$ if it is the least-dispersed distribution in \mathcal{F}_ϵ achieving a fraction $(1 - \alpha)$ of the worst case risk for $\eta(\cdot; \tau)$, i.e. if both (i)

$$\text{mse}(\nu_{\epsilon,\alpha}, \tau^\pm(\epsilon)) = (1 - \alpha) \cdot \sup_{\nu \in \mathcal{F}_\epsilon} \text{mse}(\nu, \tau^\pm(\epsilon)),$$

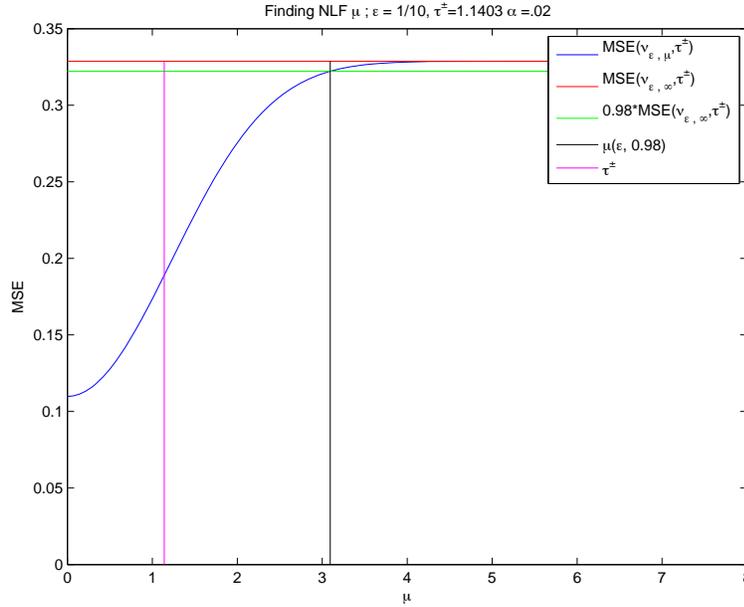


Figure 6.3: Illustration of α -least-favorable ν . For $\epsilon = 1/10$, we consider soft thresholding with the minimax parameter $\tau^\pm(\epsilon)$. We identify the smallest μ such that the measure $\nu_{\epsilon, \mu} = (1 - \epsilon)\delta_0 + \frac{\epsilon}{2}\delta_\mu + \frac{\epsilon}{2}\delta_{-\mu}$ has $\text{mse}(\nu_{\epsilon, \mu}, \tau^*) \geq 0.98 M^\pm(0.1)$ (i.e. the MSE is at least 98 % of the minimax MSE).

and (ii) ν has the smallest second moment for which (i) is true. The least favorable distribution $\nu_{\epsilon, \alpha}$ has the form of a three-point mixture

$$\nu_{\epsilon, \alpha} = (1 - \epsilon) \delta_0 + \frac{\epsilon}{2} \delta_{\mu^\pm(\epsilon, \alpha)} + \frac{\epsilon}{2} \delta_{-\mu^\pm(\epsilon, \alpha)}.$$

Here $\mu^\pm(\epsilon, \alpha)$ is an explicitly computable function, see below, and for $\alpha > 0$ fixed we have

$$\mu^\pm(\epsilon, \alpha) \sim \sqrt{2 \log(\epsilon^{-1})}, \quad \text{as } \epsilon \rightarrow 0.$$

Note in particular the relatively weak role played by α . This shows that although the precise least-favorable situation places mass at infinity, an approximately least-favorable situation is already achieved much closer to the origin.

6.3 Main results

The notation of the last section allows us to state our main results.

6.3.1 Terminology

Definition 6.3.1. (Large-System Limit). *A sequence of problem size parameters n, N will be said to **grow proportionally** if both $n, N \rightarrow \infty$ while $n/N \rightarrow \delta \in (0, 1)$.*

*Consider a sequence of random variables $(W_{n,N})$, where n, N grow proportionally. Suppose that $W_{n,N}$ converges in probability to a deterministic quantity W_∞ , which may depend on $\delta > 0$. Then we say that $W_{n,N}$ has **large-system limit** W_∞ , denoted*

$$W_\infty = \text{ls lim}(W_{n,N}).$$

Definition 6.3.2. (Large-System Framework). *We denote by $\text{LSF}(\delta, \rho, \sigma, \nu)$ a sequence of problem instances $(y, A, x^0)_{n,N}$ as per (6.1) indexed by problem sizes n, N growing proportionally: $n/N \rightarrow \delta$. In each instance, the entries of the $n \times N$ matrix A are Gaussian iid $\mathbf{N}(0, 1/n)$, the entries of z^0 are Gaussian iid $\mathbf{N}(0, \sigma^2)$ and the entries of x^0 are iid ν .*

For the sake of concreteness we focus here on problem sequences whereby the matrix A has iid Gaussian entries. An obvious generalization of this setting would be to assume that the entries are iid with mean 0 and variance $1/n$. We expect our result to hold for a broad set of distributions in this class.

In order to match the k -sparsity condition underlying (6.1) we consider the standard framework only for $\nu \in \mathcal{F}_{\delta\rho}$.

Definition 6.3.3. (Observable). *Let \hat{x} denote the output of a reconstruction algorithm on problem instance (y, A, x^0) . An observable J is a function $J(y, A, x^0, \hat{x})$ of the tuple (y, A, x^0, \hat{x}) .*

In an abuse of notation, the realized values $J_{n,N} = J(y, A, x^0, \hat{x})$ in this framework will also be called observables. An example is the observed per-coordinate MSE:

$$\text{MSE} \equiv \frac{1}{N} \|\hat{x} - x^0\|_2^2.$$

The MSE depends explicitly on x^0 and implicitly on y and A (through the reconstruction algorithm). Unless specified, we shall assume that the reconstruction algorithm solves the LASSO problem (6.2), and hence $\hat{x}^{1,\lambda} = \hat{x}$. Further in the following we will drop the dependence of the observable on the arguments y, A, x^0, \hat{x} , and the problem dimensions n, N , when clear from context.

Definition 6.3.4. (Formalism). *A formalism is a procedure that assigns a purported large-system limit $\text{Formal}(J)$ to an observable J in the $\text{LSF}(\delta, \rho, \sigma, \nu)$. This limit in general depends on δ, ρ, σ^2 , and $\nu \in \mathcal{F}_{\delta\rho}$: $\text{Formal}(J) = \text{Formal}(J; \delta, \rho, \sigma, \nu)$.*

Thus in sections below we will consider $J = \text{MSE}(y, A, x^0, \hat{x}^{1,\lambda})$ and describe a specific formalism yielding $\text{Formal}(\text{MSE})$, the formal MSE (also denoted by fmSE). Our formalism has the following character when applied to MSE: for each σ^2, δ , and probability measure ν on \mathbb{R} , it calculates a purported limit $\text{fmSE}(\delta, \nu, \sigma)$. For a problem instance with large n, N realized from the standard framework $\text{LSF}(\delta, \rho, \sigma, \nu)$, we claim the MSE will be approximately $\text{fmSE}(\delta, \nu, \sigma)$. In fact we will show how to calculate formal limits for several observables. For clarity, we always attach the modifier *formal* to any result of our formalism: e.g., *formal MSE, formal False Alarm Rate, formally optimal threshold parameter*, and so on.

Definition 6.3.5. (Validation). *A formalism is theoretically validated by proving that, in the standard asymptotic framework, we have*

$$\text{ls lim}(J_{n,N}) = \text{Formal}(J)$$

for a class \mathcal{J} of observables to which the formalism applies, and for a range of $\text{LSF}(\delta, \rho, \sigma^2, \nu)$.

A formalism is empirically validated by showing that, for problem instances (y, A, x^0) realized from $\text{LSF}(\delta, \rho, \sigma, \nu)$ with large N we have

$$J_{n,N} \approx \text{Formal}(J; \delta, \rho, \sigma, \nu),$$

for a collection of observables $J \in \mathcal{J}$ and a range of asymptotic framework parameters $(\delta, \rho, \sigma, \nu)$; here the approximation \approx should be evaluated by usual standards of

empirical science.

Obviously, theoretical validation is stronger than empirical validation, but careful empirical validation is still validation. We do not attempt here to theoretically validate this formalism in any generality; see the last chapter for the discussion in this direction. Instead we view the formalism as calculating *predictions* of empirical results. We have compared these predictions with empirical results and found a persuasive level of agreement. For example, our formalism has been used to predict the MSE of reconstructions by (6.2), and actual empirical results match the predictions, i.e.:

$$\frac{1}{N} \|\hat{x}^{1,\lambda} - x^0\|_2^2 \approx \text{fMSE}(\delta, \rho, \nu, \sigma).$$

6.3.2 Results of the formalism

The behavior of formal mean square error changes dramatically at the following phase boundary.

Definition 6.3.6 (Phase Boundary). *For each $\delta \in [0, 1]$, let $\rho_{\text{MSE}}(\delta)$ be the value of ρ solving*

$$M^\pm(\rho\delta) = \delta. \tag{6.11}$$

It is well known that $M^\pm(\epsilon)$ is monotone increasing and concave in ϵ , with $M^\pm(0) = 0$ and $M^\pm(1) = 1$. As a consequence, ρ_{MSE} is also a monotone increasing function of δ , in fact $\rho_{\text{MSE}}(\delta) \rightarrow 0$ as $\delta \rightarrow 0$ and $\rho_{\text{MSE}}(\delta) \rightarrow 1$ as $\delta \rightarrow 1$.

Proposition 6.3.7. Results of Formalism. *The formalism developed below yields the following conclusions.*

1.a *In the region $\rho < \rho_{\text{MSE}}(\delta)$, the minimax formal noise sensitivity obeys the formula*

$$M^*(\delta, \rho) \equiv \frac{M^\pm(\rho\delta)}{1 - M^\pm(\rho\delta)/\delta}.$$

In particular, M^ is finite throughout this region.*

1.b With σ^2 the noise level in (6.1), define the formal noise-plus interference level $\text{fNPI} = \text{fNPI}(\tau; \delta, \rho, \sigma, \nu)$

$$\text{fNPI} = \sigma^2 + \text{fMSE}/\delta,$$

and its minimax value $\text{NPI}^*(\delta, \rho; \sigma) \equiv \sigma^2 \cdot (1 + M^*(\delta, \rho)/\delta)$. For $\alpha > 0$, define

$$\mu^*(\delta, \rho; \alpha) \equiv \mu^\pm(\delta\rho, \alpha) \cdot \sqrt{\text{NPI}^*(\delta, \rho)}$$

In $\text{LSF}(\delta, \rho, \sigma, \nu)$ let $\nu \in \mathcal{F}_{\delta\rho}$ place fraction $1 - \delta\rho$ of its mass at zero and the remaining mass equally on $\pm\mu^*(\delta, \rho; \alpha)$. This ν is $\tilde{\alpha}$ -least-favorable: the formal noise sensitivity of $\hat{x}^{1,\lambda}$ equals $(1 - \tilde{\alpha})M^*(\delta, \rho)$, with $(1 - \tilde{\alpha}) = (1 - \alpha)(1 - M^\pm(\delta\rho))/(1 - (1 - \alpha)M^\pm(\delta\rho))$.

1.c The formally maximin penalty parameter obeys

$$\lambda^*(\nu; \delta, \rho, \sigma) \equiv \tau^\pm(\delta\rho) \cdot \sqrt{\text{fNPI}(\tau^\pm; \delta, \rho, \sigma, \nu)} \cdot (1 - \text{EqDR}(\nu; \tau^\pm(\delta\rho))/\delta),$$

where $\text{EqDR}(\dots)$ is the asymptotic detection rate, i.e. the asymptotic fraction of coordinates that are estimated to be nonzero. (An explicit expression for this quantity is given in Section 6.16.)

In particular with this ν -adaptive choice of penalty parameter, the formal MSE of $\hat{x}^{1,\lambda}$ does not exceed $M^* \cdot \sigma^2$.

2 In the region $\rho > \rho_{\text{MSE}}(\delta)$, the formal noise sensitivity is infinite. Throughout this phase, for each fixed number $M < \infty$, there exists $\alpha > 0$ such that the probability distribution $\nu \in \mathcal{F}_{\delta\rho}$ placing its nonzeros at $\pm\mu^*(\delta, \rho, \alpha)$, yields formal MSE larger than M .

We explain the formalism and derive these results in Section 6.4 below.

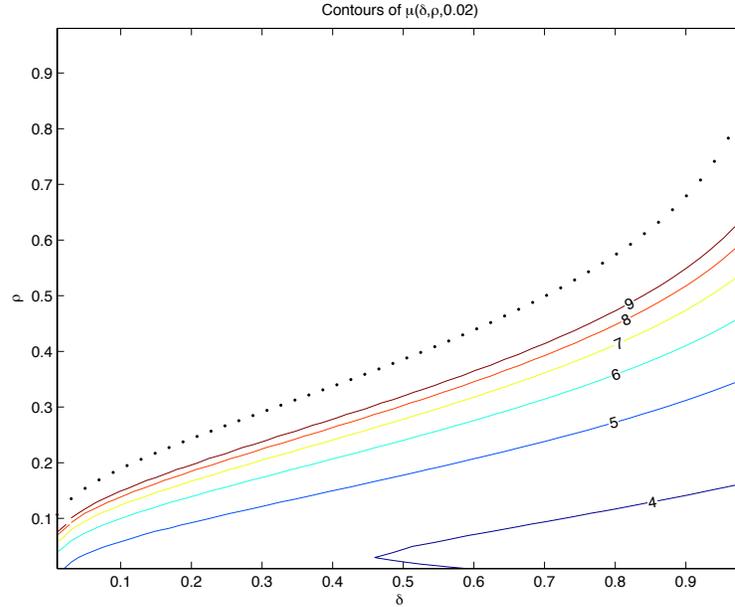


Figure 6.4: Contour lines of the near-least-favorable signal amplitude $\mu^*(\delta, \rho, \alpha)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while the colored solid lines portray level sets of $\mu^*(\delta, \rho, \alpha)$. The 3-point mixture distribution $(1 - \epsilon)\delta_0 + \frac{\epsilon}{2}\delta_\mu + \frac{\epsilon}{2}\delta_{-\mu}$, $(\epsilon = \delta\rho)$ will cause 98% of the worst-case MSE. When a k -sparse vector is drawn from this distribution, its nonzeros are all at $\pm\mu$.

6.3.3 Interpretation of the predictions

Figure 6.1 displays the noise sensitivity; above the phase transition boundary $\rho = \rho_{\text{MSE}}(\delta)$, it is infinite. The different contour lines show positions in the δ, ρ plane where a given noise sensitivity is achieved. As one might expect, the sensitivity blows up rather dramatically as we approach the phase boundary.

Figure 6.4 displays the least-favorable coefficient amplitude $\mu^*(\delta, \rho, \alpha = 0.02)$. Notice that $\mu^*(\delta, \rho, \alpha)$ diverges as the phase boundary is approached. Indeed beyond the phase boundary arbitrarily large MSE can be produced by choosing μ large enough.

Figure 6.5 displays the value of the optimal penalization parameter amplitude $\lambda^* = \lambda^*(\nu_{\delta, \rho}^*; \delta, \rho, \sigma = 1)$. Note that the parameter tends to zero as we approach phase transition.

For these figures, the region above phase transition is not decorated, because the values there are infinite or not defined.

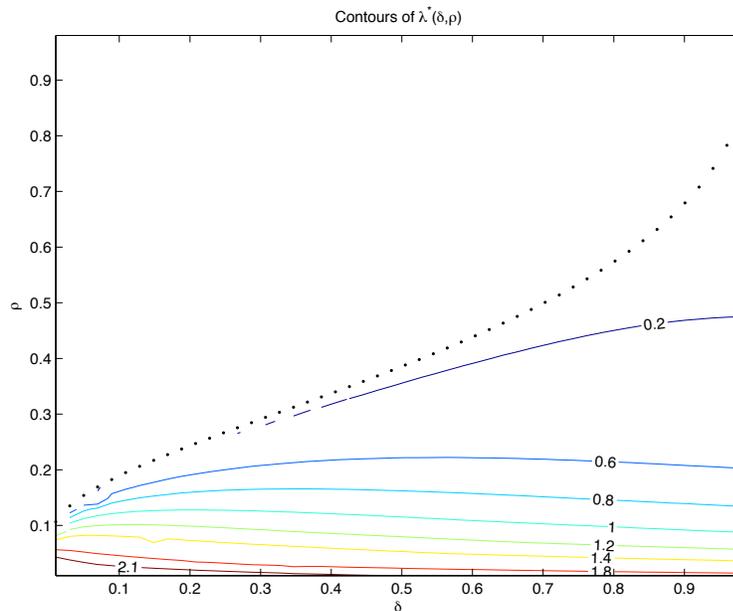


Figure 6.5: Contour lines of the maximin penalization parameter: $\lambda^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while thin lines are contours for $\lambda^*(\delta, \rho, \alpha)$. Close to phase transition, the maximin value approaches 0.

6.3.4 Comparison to other phase transitions

In view of the importance of the phase boundary for Proposition 6.3.7, we note the following:

Finding 3. Phase Boundary Equivalence. *The phase boundary ρ_{MSE} is identical to the phase boundary ρ_{ℓ_1} below which ℓ_1 minimization and ℓ_0 minimization are equivalent.*

In words, throughout the phase where ℓ_1 minimization is equivalent to ℓ_0 minimization, the solution to (6.2) has bounded formal MSE. When we are outside that phase, the solution has unbounded formal MSE. The verification of Finding 3 follows in two steps. First, the formulas for the phase boundary discussed in this chapter are identical to the phase boundary formulas given in chapter 4; Second, in that chapter it was shown that these formulas agree numerically with the formulas known for ρ_{ℓ_1} .

6.3.5 Validating the predictions

Proposition 6.3.7 makes predictions for the behavior of solutions to (6.2). It will be validated empirically, by showing that such solutions behave as predicted.

In particular, simulation evidence will be presented to show that in the phase where noise sensitivity is *finite*:

1. Running (6.2) for data (y, A) generated from vectors x_0 with coordinates with distribution ν which is nearly least-favorable results in an empirical MSE approximately equal to $M^*(\delta, \rho) \cdot \sigma^2$.
2. Running (6.2) for data (y, A) generated from vectors x_0 with coordinates with distribution ν which is far from least-favorable results in empirical MSE noticeably smaller than $M^*(\delta, \rho) \cdot \sigma^2$.
3. Running (6.2) with a suboptimal penalty parameter λ results in empirical MSE noticeably greater than $M^*(\delta, \rho) \cdot \sigma^2$.

Second, in the phase where formal MSE is *infinite*:

4. Running (6.2) on vectors x_0 generated by formally least-favorable results in an empirical MSE which is very large.

Evidence for all these claims will be given below.

6.4 The formalism

The formalism that is used for deriving the above properties is the state evolution recursions that we derived for the AMP algorithm. For the sake of completeness of this chapter we summarize our discussion of previous chapter here.

6.4.1 The AMPT algorithm

We now consider *first-order approximate message passing* (AMP) algorithm reconstruction algorithm. Starting at $\hat{x}^0 = 0$ it proceeds iteratively and produces the

estimate \hat{x}^t of x^0 at iteration t according to the iteration:

$$z^t = y - A\hat{x}^t + z^{t-1} \frac{df_t}{n} \quad (6.12)$$

$$\hat{x}^{t+1} = \eta(A^*z^t + \hat{x}^t; \theta_t), \quad (6.13)$$

Here $\hat{x}^t \in \mathbb{R}^p$ is the current estimate of x^0 , and $df_t = \|\hat{x}^t\|_0$ is the number of nonzeros in the current estimate. Again $\eta(\cdot; \cdot)$ is the *soft threshold* nonlinearity with threshold parameter θ_t

$$\theta_t = \tau \cdot \sigma_t; \quad (6.14)$$

τ is a tuning constant, fixed throughout iterations and σ_t is an empirical measure of the scale of the residuals. Finally $z^t \in \mathbb{R}^n$ is the current *working residual*. Compare with the usual residual defined by $r^t = y - A\hat{x}^t$ via the identity $z^t = r^t + z^{t-1} \frac{df_t}{n}$. The extra term in AMP plays a subtle but crucial role.¹

6.4.2 Formal MSE, and its evolution

Let $\text{npi}(m; \sigma, \delta) \equiv \sigma^2 + m/\delta$. We define the *MSE map* Ψ through

$$\Psi(m, \delta, \sigma, \tau, \nu) \equiv \text{mse}(\text{npi}(m, \sigma, \delta); \nu, \tau), \quad (6.15)$$

where the function $\text{mse}(\cdot; \nu, \tau)$ is the soft thresholding mean square error already introduced in (6.7). It describes the MSE of soft thresholding in a problem where the noise level is $\sqrt{\text{npi}}$. A heuristic explanation of the meaning and origin of npi will be given below.

¹The only difference between this algorithm and the AMPM algorithm introduced in the last chapter is the choice of threshold; instead of a tuning parameter τ like in (6.14) – one that can be set freely – a fixed choice $\tau(\delta)$ was made for each specific δ through the maximin framework. Here we call that algorithm AMPM - *M* for *maximin*. In contrast, the current algorithm is tunable, allowing choice of τ , we label it AMPT(τ), *T* for tunable.

Definition 6.4.1. State Evolution. *The state is a 5-tuple $(m; \delta, \sigma, \tau, \nu)$. State evolution is the evolution of the state by the rule*

$$\begin{aligned} (m_t; \delta, \sigma, \tau, \nu) &\mapsto (\Psi(m_t); \delta, \sigma, \tau, \nu), \\ t &\mapsto t + 1. \end{aligned}$$

As the parameters $(\delta, \sigma, \tau, \nu)$ remain fixed during evolution, we usually omit mention of them and think of state evolution simply as the iterated application of Ψ :

$$\begin{aligned} m_t &\mapsto m_{t+1} \equiv \Psi(m_t), \\ t &\mapsto t + 1. \end{aligned}$$

Definition 6.4.2. Stable Fixed Point. *The Highest Fixed Point of the continuous function Ψ is*

$$\text{HFP}(\Psi) = \sup\{m : \Psi(m) \geq m\}.$$

The stability coefficient of the continuously differentiable function Ψ is

$$\text{SC}(\Psi) = \left. \frac{d}{dm} \Psi(m) \right|_{m=\text{HFP}(\Psi)}.$$

We say that $\text{HFP}(\Psi)$ is a stable fixed point if $0 \leq \text{SC}(\Psi) < 1$.

To illustrate this, Figure 6.6 shows the MSE map and fixed points in three cases.

In what follows we denote by $\mu_2(\nu) = \int x^2 d\nu$ the second-moment of the distribution ν .

Lemma 6.4.1. *Let $\Psi(\cdot) = \Psi(\cdot, \delta, \sigma, \tau, \nu)$, and assume either $\sigma^2 > 0$ or $\mu_2(\nu) > 0$. Then the sequence of iterates m_t defined by $m_{t+1} = \Psi(m_t)$ starting from $m_0 = \mu_2(\nu)$ converges monotonically to $\text{HFP}(\Psi)$:*

$$m_t \rightarrow \text{HFP}(\Psi), \quad t \rightarrow \infty.$$

Further, if $\sigma > 0$ then $\text{HFP}(\Psi) \in (0, \infty)$ is the unique fixed point.

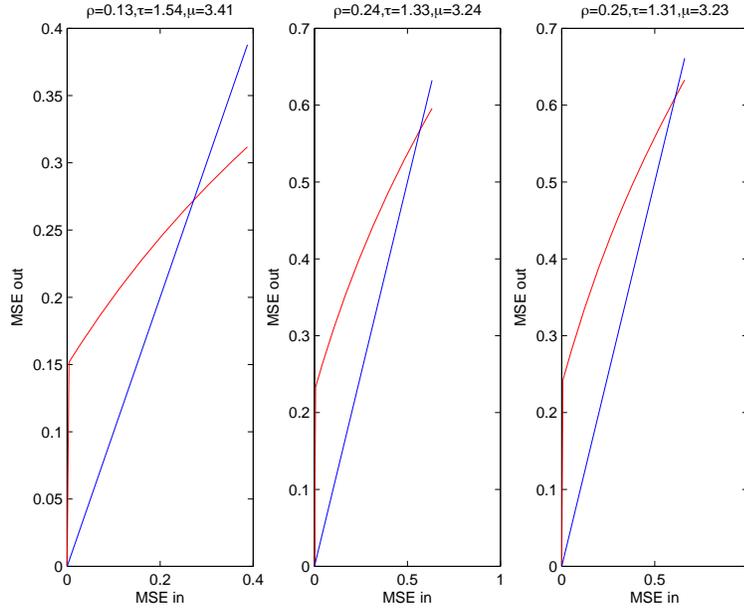


Figure 6.6: MSE Map Ψ in three cases, and associated fixed points. Left: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}/2$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$ Center: $\delta = 0.25$, $\rho = \rho_{\text{MSE}} \times 0.95$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$ Right: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu^*(\delta, \rho, \alpha)$

Suppose further that the stability coefficient satisfies $0 < \text{SC}(\Psi) < 1$. Then there exists a constant $\mathcal{A}(\nu, \Psi)$ such that

$$|m_t - \text{HFP}(\Psi)| \leq \mathcal{A}(\nu, \Psi) \text{SC}(\Psi)^t.$$

Finally, if $\mu_2(\nu) \geq \text{HFP}(\Psi)$ then the sequence $\{m_t\}$ is monotonically decreasing to $\mu_2(\nu)$ with

$$(m_t - \text{HFP}(\Psi)) \leq \text{SC}(\Psi)^t \cdot (\mu_2(\nu) - \text{HFP}(\Psi)).$$

In short, barring the trivial case $x^0 = 0$, $z^0 = 0$ (no signal, no noise), state evolution converges to the highest fixed point. If the stability coefficient is smaller than 1, convergence is exponentially fast.

Proof (Lemma 6.4.1). This lemma is an immediate consequence of the fact that $m \mapsto \Psi(m)$ is a concave non-decreasing function, with $\Psi(0) > 0$ as long as $\sigma > 0$ and $\Psi(0) = 0$ for $\sigma = 0$.

Indeed in the last chapter it was shown that at noise level $\sigma = 0$, the MSE map $m \rightarrow \Psi(m; \delta, \sigma, \nu, \tau)$ is concave as a function of m . We have the identity

$$\Psi(m; \delta, \sigma, \nu, \tau) = \Psi(m + \sigma^2 \cdot \delta; \delta, \sigma = 0, \nu, \tau),$$

relating the noise-level 0 MSE map to the noise-level σ MSE map. From this it follows that Ψ is concave for $\sigma > 0$ as well. Also, we showed that $\Psi(m = 0; \delta, \sigma = 0, \nu, \tau) = 0$ and $\frac{d\Psi}{dm}(m = 0; \delta, \sigma = 0, \nu, \tau) > 0$, whence $\Psi(m = 0; \delta, \sigma, \nu, \tau) > 0$ for any positive noise level σ . \square

Lemma 6.4.2. *If $M^\pm(\delta, \rho) < \delta$ in the noiseless case $\sigma = 0$, the only fixed point of the Ψ is zero. Furthermore*

$$\text{SC}^*(\delta, \rho, \sigma = 0) \triangleq \inf_{\tau} \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{SC}(\Psi(\cdot; \delta, \sigma = 0, \nu, \tau)) = \frac{M^\pm(\delta\rho)}{\delta}.$$

Proof.

$$\begin{aligned} \mathbb{E}(\eta(X + \sqrt{\frac{m}{\delta}}Z; \tau\sqrt{\frac{m}{\delta}}) - X) &= \frac{m}{\delta} \mathbb{E}(\eta(\sqrt{\delta/m}X + Z; \tau) - \sqrt{\delta/m}X) \\ &\leq \sup_{\nu \in \mathcal{F}_{\delta\rho}} \mathbb{E}(\eta(X + \sqrt{\frac{m}{\delta}}Z; \tau\sqrt{\frac{m}{\delta}}) - X). \end{aligned}$$

By taking the infimum over τ we have,

$$\begin{aligned} \inf_{\tau} \mathbb{E}(\eta(X + \sqrt{\frac{m}{\delta}}Z; \tau\sqrt{\frac{m}{\delta}}) - X) &\leq \inf_{\tau} \sup_{\nu \in \mathcal{F}_{\delta\rho}} \mathbb{E}(\eta(X + \sqrt{\frac{m}{\delta}}Z; \tau\sqrt{\frac{m}{\delta}}) - X) \\ &\leq \frac{M^\pm(\delta\rho)}{\delta} m. \end{aligned}$$

Therefore if $\frac{M^\pm(\delta\rho)}{\delta} < 1$ then the only fixed point of Ψ is $m = 0$. In the previous chapter we showed that at $m = 0$ the derivative of Ψ is independent of the distribution of the non-zero elements. Also since $\Psi(0) = 0$ we have,

$$\frac{\Psi(m) - \Psi(0)}{m} \leq \frac{M^\pm(\delta\rho)}{\delta}.$$

By taking the limit $m \rightarrow 0$ we get $\text{SC}^*(\delta, \rho, \sigma = 0) \leq \frac{M^\pm(\delta\rho)}{\delta}$. It can also be proved that if we calculate $\Psi(m)$ on the least favorable distribution then we have $\Psi(m) = \frac{M^\pm(\delta\rho)}{\delta}m$ which will prove the equality². \square

For this reason, the region $M^\pm(\delta, \rho) < \delta$ can also be called the *stability phase*, not only the stability coefficient is smaller than 1, $\text{SC}(\Psi) < 1$, but that it can be bounded away from 1 uniformly in the signal distribution ν . Outside the stability region, for each large m , we can find measures ν obeying the sparsity constraint $\nu \in \mathcal{F}_{\delta\rho}$ for which state evolution converges to a fixed point suffering equilibrium MSE $> m$. The construction in Section 6.4.5 shows that $\text{HFP}(\Psi) > \mu_2(\nu) > m$. Figure 6.7 shows the MSE map and the state evolution in three cases which may be compared to 6.6. In the first case, ρ is well below ρ_{MSE} and the fixed point is well below $\mu_2(\nu)$. In the second case, ρ is slightly below ρ_{MSE} and the fixed point is close to $\mu_2(\nu)$. In the third case, ρ is above ρ_{MSE} and the fixed point, lies above $\mu_2(\nu)$.

$\mu_2(\nu)$ is the MSE one suffers by ‘doing nothing’: setting threshold $\lambda = \infty$ and taking $\hat{x} = 0$. When $\text{HFP}(\Psi) > \mu_2(\nu)$, one iteration of thresholding makes things *worse*, not better. In words, the phase boundary is exactly the place below which we are sure that, if $\mu_2(\nu)$ is large, a single iteration of thresholding gives an estimate \hat{x}^1 that is better than the starting point \hat{x}^0 . Above the phase boundary, even a single iteration of thresholding may be a catastrophically bad thing to do.

Definition 6.4.3. (Equilibrium States and State-Conditional Expectations)

Consider a real-valued function $\zeta : \mathbb{R}^3 \mapsto \mathbb{R}$, its expectation in state $S = (m; \delta, \sigma, \nu)$ is

$$\mathcal{E}(\zeta|S) = \mathbb{E} \left\{ \zeta(X, Z, \eta(X + \sqrt{\text{npi}} Z; \tau\sqrt{\text{npi}})) \right\},$$

where $\text{npi} = \text{npi}(m; \sigma, \delta)$ and $X \sim \nu$, $Z \sim \mathbf{N}(0, 1)$ are independent random variables.

Suppose we are given $(\delta, \sigma, \nu, \tau)$, and a fixed point m^* , $m^* = \text{HFP}(\Psi)$ with $\Psi = \Psi(\cdot; \delta, \sigma, \nu, \tau)$. The tuple $S^* = (m^*; \delta, \sigma, \nu)$ is called the equilibrium state of state evolution. The expectation in the equilibrium state is $\mathcal{E}(\zeta|S^*)$.

²Since the least favorable distribution has a point mass at ∞ we should make this statement precise by a limit argument

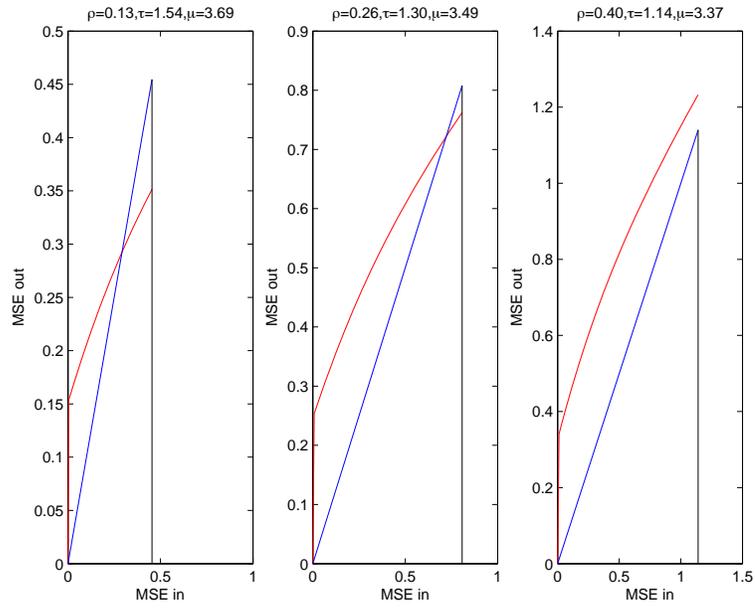


Figure 6.7: Crossing the phase transition: effects on MSE Map Ψ , and associated state evolution. Left: $\delta = 0.25$, $\rho = \rho_{\text{MSE}}/2$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$ Middle: $\delta = 0.25$, $\rho = 0.9 \cdot \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$ Right: $\delta = 0.25$, $\rho = 1.5 \cdot \rho_{\text{MSE}}$, $\sigma = 1$, $\nu = \nu(\delta, \rho, 0.01)$. In each case $\tau = \tau^\pm(\delta\rho)$.

Name	Abbrev.	$\zeta = \zeta(u, v, w)$
Mean Square Error	MSE	$\zeta = (u - w)^2$
False Alarm Rate	FAR	$\zeta = 1_{\{w \neq 0 \& u = 0\}} / (1 - \rho\delta)$
Detection Rate	DR	$\zeta = 1_{\{w \neq 0\}}$
Missed Detection Rate	MDR	$\zeta = 1_{\{w = 0 \& u \neq 0\}} / (\rho\delta)$
False Detection Rate	FDeR	$\zeta = 1_{\{w \neq 0 \& u = 0\}} / (\rho\delta)$

Table 6.1: Some observables and their names.

Definition 6.4.4. (State Evolution Formalism for AMPT) . Run the AMPT algorithm and assume that the sequence of estimates (\hat{x}^t, z^t) converges to the fixed point $(\hat{x}^\infty, z^\infty)$. To each function $\zeta : \mathbb{R}^3 \mapsto \mathbb{R}$ associate the observable

$$J^\zeta(y, A, x^0, \hat{x}) = \frac{1}{N} \sum_{i=1}^N \zeta(x^0(i), A^T z(i) + \hat{x}(i) - x^0(i), \hat{x}(i)).$$

Let S^* denote the equilibrium state reached by state evolution in a given situation $(\delta, \sigma, \nu, \tau)$. The state evolution formalism assigns the purported limit value

$$\text{Formal}(J^\zeta) = \mathcal{E}(\zeta|S^*).$$

Validity of the state evolution formalism for AMPT entails that, for a sequence of problem instances (y, A, x^0) drawn from $\text{LSF}(\delta, \rho, \sigma, \nu)$, the large-system limit for observable $J_{n,N}^\zeta$ is simply the expectation in the equilibrium state:

$$\text{ls lim } J_{n,N}^\zeta = \mathcal{E}(\zeta|S^*).$$

The class \mathcal{J} of observables representable by the form J^ζ is quite rich, by choosing $\zeta(u, v, w)$ appropriately. Table 6.1 gives examples of well-known observables and the ζ which will generate them. Formal values for other interesting observables can in principle be obtained by combining such simple ones. For example, the False Discovery rate FDR is the ratio FDeR/DR and so the ratio of two elementary observables

of the kind for which the formalism is defined. We assign it the purported limit value

$$\text{Formal(FDR)} = \frac{\text{Formal(FDeR)}}{\text{Formal(DR)}}.$$

Below we list a certain number of observables for which the formalism was checked empirically and that play an important role in characterizing the fixed point estimates.

Calculation of Formal Operating Characteristics of AMPT(τ) by State Evolution

Given $\delta, \sigma, \nu, \tau$, identify the fixed point $\text{HFP}(\Psi(\cdot; \delta, \sigma, \nu, \tau))$. Calculate the following quantities

- Equilibrium MSE

$$\text{EqMSE} = m_\infty = \text{HFP}(\Psi(\cdot; \nu, \tau); \delta, \sigma).$$

- Equilibrium Noise Plus Interference Level

$$\text{np}_\infty = \frac{1}{\delta} m_\infty + \sigma^2$$

- Equilibrium Threshold (absolute units)

$$\theta_\infty = \tau \cdot \sqrt{\text{np}_\infty}.$$

- Equilibrium Mean Squared Residual. Let $Y_\infty = X + \sqrt{\text{np}_\infty} Z$ for $X \sim \nu$ and $Z \sim \text{N}(0, 1)$ are independent. Then

$$\text{EqMSR} = \mathbb{E}\{ [Y_\infty - \eta(Y_\infty; \theta_\infty)]^2 \}.$$

- Equilibrium Mean Absolute Estimate

$$\text{EqMAE} = \mathbb{E}\{ |\eta(Y_\infty; \theta_\infty)| \}.$$

– Equilibrium Detection Rate

$$\text{EqDR} = \mathbb{P}\{\eta(Y_\infty; \theta_\infty) \neq 0\}. \quad (6.16)$$

– Equilibrium Penalized MSR

$$\text{EqPMSR} = \text{EqMSR}/2 + \theta_\infty \cdot (1 - \text{EqDR}/\delta) \cdot \text{EqMAE}.$$

6.4.3 AMPT-LASSO calibration

Of course at this point the reader is entitled to feel that the introduction of AMPT is a massive digression. The relevance of AMPT is indicated by the following conclusion:

Finding 4. *In the large system limit, the operating characteristics of AMPT(τ) are equivalent to those of LASSO(λ) under an appropriate calibration $\tau \leftrightarrow \lambda$.*

By *calibration*, we mean a rescaling that maps results on one problem into results on the other problem. The correct mapping can be guessed from the following remarks:

LASSO(λ): no *residual* exceeds λ : $\|A^T(y - A\hat{x}^{1,\lambda})\|_\infty \leq \lambda$. Further

$$\begin{aligned} \hat{x}_i^{1,\lambda} > 0 &\Leftrightarrow (A^T(y - A\hat{x}^{1,\lambda}))_i = \lambda, \\ \hat{x}_i^{1,\lambda} = 0 &\Leftrightarrow |(A^T(y - A\hat{x}^{1,\lambda}))_i| < \lambda, \\ \hat{x}_i^{1,\lambda} < 0 &\Leftrightarrow (A^T(y - A\hat{x}^{1,\lambda}))_i = -\lambda. \end{aligned}$$

- AMPT(τ): At a fixed point \hat{x}^∞, z^∞ , no *working residual* exceeds the equilibrium threshold θ_∞ : $\|A^T z^\infty\|_\infty \leq \theta_\infty$. Further

$$\begin{aligned} \hat{x}_i^\infty > 0 &\Leftrightarrow (A^T z^\infty)_i = \theta_\infty, \\ \hat{x}_i^\infty = 0 &\Leftrightarrow |(A^T z^\infty)_i| < \theta_\infty, \\ \hat{x}_i^\infty < 0 &\Leftrightarrow (A^T z^\infty)_i = -\theta_\infty. \end{aligned}$$

Define $df = \#\{i : \hat{x}_i^\infty \neq 0\}$. Further notice that at the AMPT fixed point $(1 - df/n)z^\infty = y - A^T \hat{x}^\infty$. We can summarize these remarks in the following statement.

Lemma 6.4.3. *Solutions $\hat{x}^{1,\lambda}$ of $\text{LASSO}(\lambda)$ (i.e. optima of the problem (6.2)) are in correspondence with fixed points $(\hat{x}^\infty, z^\infty)$ of the $\text{AMPT}(\tau)$ under the bijection $\hat{x}^\infty = \hat{x}^{1,\lambda}$, $z^\infty = (y - A^T \hat{x}^{1,\lambda})/(1 - df/n)$, provided the threshold parameters are in the following relation*

$$\lambda = \theta_\infty \cdot (1 - df/n). \quad (6.17)$$

In other words, if we have a fixed point of $\text{AMPT}(\tau)$ we can choose λ in such a way that this is also an optimum of $\text{LASSO}(\lambda)$. Viceversa, any optimum of $\text{LASSO}(\lambda)$ can be realized as a fixed point of $\text{AMPT}(\tau)$: notice in fact that the relation (6.17) is invertible whenever $df < n$.

This simple rule gives a calibration relationship between τ and λ , i.e. a one-one correspondence between τ and λ that renders the two apparently different reconstruction procedures equivalent, provided the iteration $\text{AMPT}(\tau)$ converges rapidly to its fixed point. Our empirical results confirm that this is indeed what happens for typical large system frameworks $\text{LSF}(\delta, \rho, \sigma, \nu)$.

The next lemma characterizes the equilibrium calibration relation between AMP and LASSO.

Lemma 6.4.4. *Let $\text{EqDR}(\tau) = \text{EqDR}(\tau; \delta, \rho, \nu, \sigma)$ denote the equilibrium detection rate obtained from state evolution when the tuning parameter of AMPT is τ . Define $\tau^0(\delta, \rho, \nu, \sigma) > 0$, so that $\text{EqDR}(\tau) \leq \delta$ when $\tau > \tau^0$. For each $\lambda \geq 0$, there is a unique value $\tau(\lambda) \in [\tau_0, \infty)$ such that*

$$\lambda = \theta_\infty(\tau) \cdot (1 - \text{EqDR}(\tau)/\delta).$$

We can restate Finding 4 in the following more convenient form.

Finding 5. *For each $\lambda \in [0, \infty)$ we find that $\text{AMPT}(\tau(\lambda))$ and $\text{LASSO}(\lambda)$ have statistically equivalent observables. In particular the MSE, MAE, MSR, DR, have the same distributions.*

6.4.4 Derivation of proposition 6.3.7

Consider the following minimax problem for $AMPT(\tau)$. With $\text{fmSE}(\tau; \delta, \rho, \sigma, \nu)$ denoting the equilibrium formal MSE for $AMPT(\tau)$ for the framework $\text{LSF}(\delta, \rho, \sigma, \nu)$, fix $\sigma = 1$ and *define*

$$M^b(\delta, \rho) = \inf_{\tau} \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fmSE}(\tau; \delta, \rho, \sigma = 1, \nu). \quad (6.18)$$

We will first show that this definition obeys the formula just like the one in Proposition 6.3.7, given for M^* . Later we show that $M^b = M^*$.

Proposition 6.4.5. *For M^b defined by (6.18),*

$$M^b(\delta, \rho) = \frac{M^\pm(\delta\rho)}{1 - M^\pm(\delta\rho)/\delta} \quad (6.19)$$

The AMPT threshold rule

$$\tau^*(\delta, \rho) = \tau^\pm(\delta\rho), \quad 0 < \rho < \rho_{\text{MSE}}(\delta), \quad (6.20)$$

minimizes the formal MSE:

$$\sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fmSE}(\tau^*; \delta, \rho, 1, \nu) = \inf_{\tau} \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fmSE}(\tau; \delta, \rho, 1, \nu) = M^b(\delta, \rho). \quad (6.21)$$

Figure 6.8 depicts the behavior of τ^* in the (δ, ρ) plane.

Proof. For the simplicity of notation we consider m_∞ instead of $M^b(\delta, \rho)$. First suppose that m_∞ satisfies the following equation,

$$m_\infty = \inf_{\tau} \sup_{\mathcal{F}_{\delta\rho}} \mathbb{E}(\eta(X + \sqrt{\sigma^2 + m_\infty/\delta}Z; \tau\sqrt{\sigma^2 + m_\infty/\delta}) - X)^2. \quad (6.22)$$

If we prove this then the statement of the theorem can be easily derived from above. We know that if the distribution of X belongs to $\mathcal{F}_{\delta\rho}$ then the distribution of aX

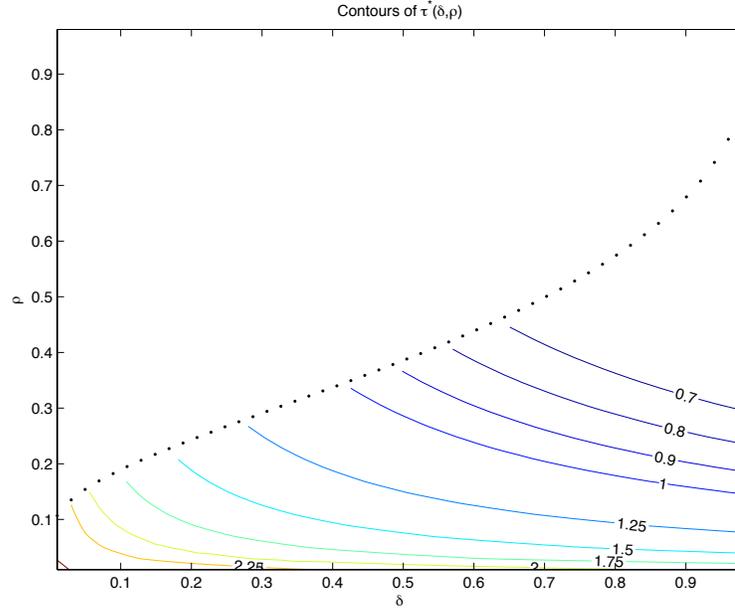


Figure 6.8: Contour lines of $\tau^*(\delta, \rho)$ in the (ρ, δ) plane. The dotted line corresponds to the phase transition $(\delta, \rho_{\text{MSE}}(\delta))$, while thin lines are contours for $\tau^*(\delta, \rho)$

clearly belongs to $\mathcal{F}_{\delta\rho}$ and therefore

$$m_\infty = (\sigma^2 + m_\infty/\delta) \inf_{\tau} \sup_{\mathcal{F}_{\delta\rho}} \mathbb{E}(\eta(X + Z; \tau) - X)^2 = (\sigma^2 + m_\infty/\delta) M^*(\delta\rho).$$

which is equivalent to the statement of the theorem. Therefore we should just prove (6.22). Suppose that τ^* and ν^* correspond to the saddle point of (6.18)³ and the fixed point is called m^* . Then we have,

$$m^* = \mathbb{E}(\eta(X + \sqrt{\sigma^2 + m^*/\delta}Z; \tau^* \sqrt{\sigma^2 + m^*/\delta}) - X)^2.$$

The goal here is to prove that (τ^*, ν^*) plays the role of the saddle point for (6.22). We prove this by contradiction. If this is not true, one of these two cases may happen. Either there exist ν_0 for which τ_0 is the minimizer of right hand side of (6.22) and this ν_0 is able to increase the RHS of (6.22) or there exist a τ_0 and the corresponding

³Here for simplicity of the notation we assume that the inf and sup are achieved. By limit arguments these results can be extended to other cases.

maximizing distribution ν_0 such that RHS of (6.22) is smaller. Since the argument for both cases is very similar we just prove that the first thing can not happen. If the first scenario happens this means that,

$$m_\infty < \mathbb{E}_{X \sim \nu_0} (\eta(X + \sqrt{(\sigma^2 + m_\infty/\delta)}Z; \tau_0 \sqrt{(\sigma^2 + m_\infty/\delta)}) - X)^2.$$

Therefore there will be another fixed point larger than m_∞ which is in contradiction with the fact that τ^*, ν^* is the saddle point of (6.18). Since the saddle points are the same the two equations become equivalent. \square

We now explain how this result about AMPT leads to our claim for the behavior of the LASSO estimator $\hat{x}^{1,\lambda}$. By a scale invariance the quantity (6.3) can be rewritten as a fixed-scale $\sigma = 1$ property:

$$M^*(\delta, \rho) = \sup_{\nu \in \mathcal{F}_{\delta\rho}} \inf_{\lambda} \text{fmSE}(\nu, \lambda | \text{LASSO}),$$

where we introduced explicit reference to the algorithm used, and dropped the irrelevant arguments. We will analogously write $\text{fmSE}(\nu, \tau | \text{AMPT})$ for the AMPT(τ) MSE.

Proposition 6.4.6. *Assume the validity of our calibration relation i.e. the equivalence of formal operating characteristics of AMPT(τ) and LASSO($\lambda(\tau)$). Then*

$$M^*(\delta, \rho) = M^b(\delta, \rho).$$

Also, for λ^* as defined in Proposition 6.3.7,

$$M^*(\delta, \rho) = \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fmSE}(\nu, \lambda^*(\nu; \delta, \rho, \sigma) | \text{LASSO}).$$

In words, λ^* is the maximin penalization and the maximin MSE of LASSO is precisely given by the formula (6.19).

Proof. Taking the validity of our calibration relationship $\tau \leftrightarrow \lambda(\tau)$ as given, we must

have

$$\text{fMSE}(\nu, \lambda(\tau)|\text{LASSO}) = \text{fMSE}(\nu, \tau|\text{AMPT}).$$

Our definition of λ^* in Proposition 6.3.7 is simply the calibration relation applied to the minimax AMPT threshold τ^* , i.e. $\lambda^* = \lambda(\tau^*)$. Hence assuming the validity of our calibration relation, we have:

$$\begin{aligned} \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fMSE}(\nu, \lambda^*(\nu; \delta, \rho, \sigma)|\text{LASSO}) &= \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fMSE}(\nu, \lambda(\tau^*)|\text{LASSO}) \\ &= \sup_{\nu \in \mathcal{F}_{\delta\rho}} \text{fMSE}(\nu, \tau^*|\text{AMPT}) \\ &= \sup_{\nu \in \mathcal{F}_{\delta\rho}} \inf_{\tau} \text{fMSE}(\nu, \tau|\text{AMPT}) \quad (6.23) \\ &= \sup_{\nu \in \mathcal{F}_{\delta\rho}} \inf_{\tau} \text{fMSE}(\nu, \lambda(\tau)|\text{LASSO}) \\ &= \sup_{\nu \in \mathcal{F}_{\delta\rho}} \inf_{\lambda} \text{fMSE}(\nu, \lambda|\text{LASSO}). \end{aligned}$$

Display (6.23) shows that all these equalities are equal to $M^b(\delta, \rho)$. \square

The proof of Proposition 6.3.7, points 1a, 1b, 1c follows immediately from the above.

6.4.5 Formal MSE above phase transition

We now make an explicit construction showing that noise sensitivity is unbounded above PT.

We first consider the AMPT algorithm above PT. Fix δ, ρ with $\rho > \rho_{\text{MSE}}(\delta)$ and set $\epsilon = \delta\rho$.

In this section we focus on 3 point distributions with mass at 0 equal to $1 - \epsilon$. With an abuse of notation we let $\text{mse}(\mu, \tau)$ denote the MSE of scalar soft thresholding for amplitude of the non-zeros equal to μ , and noise variance equal to 1. In formulas, $\text{mse}(\mu, \tau) \equiv \text{mse}(1; (1 - \epsilon)\delta_0 + (\epsilon/2)\delta_\mu + (\epsilon/2)\delta_{-\mu}, \tau)$, and

$$\text{mse}(\mu, \tau) = (1 - \epsilon)\mathbb{E}\eta(Z; \tau)^2 + \epsilon\mathbb{E}(\mu - \eta(\mu + Z; \tau))^2.$$

Consider values of the AMPT threshold τ such that $\text{mse}(0, \tau) < \delta$; this will be possible for all τ sufficiently large. Pick a number $\gamma \in (0, 1)$ obeying

$$1 < \gamma < \text{mse}(0, \tau)/\delta. \quad (6.24)$$

Let $M^\pm(\epsilon, \tau) = \sup_\mu \text{mse}(\mu, \tau)$ denote the worst case risk of $\eta(\cdot; \tau)$ over the class \mathcal{F}_ϵ . Let $\mu^\pm(\epsilon, \alpha, \tau)$ denote the α -least-favorable μ for threshold τ :

$$\text{mse}(\mu^\pm, \tau) = (1 - \alpha)M^\pm(\epsilon, \tau).$$

Define $\alpha^* = 1 - \gamma\delta/M^\pm(\epsilon, \tau)$, and note that $\alpha^* \in (0, 1)$ by earlier assumptions. Let $\mu^* = \mu^\pm(\alpha^*, \tau, \epsilon)$. A straightforward calculation along the lines of the previous section yields.

Lemma 6.4.5. *For the measure $\nu = (1 - \epsilon)\delta_0 + (\epsilon/2)\delta_{\mu^*} + (\epsilon/2)\delta_{-\mu^*}$, the formal MSE and formal NPI are given by*

$$\begin{aligned} \text{fMSE}(\nu, \tau | \text{AMPT}) &= \frac{\delta\gamma}{1 - \gamma}, \\ \text{fNPI}(\nu, \tau | \text{AMPT}) &= \frac{1}{1 - \gamma}. \end{aligned}$$

Assumption (6.24) permits us to choose γ very close to 1. Hence the above formulas show explicitly that MSE is unbounded above phase transition.

What do the formulas say about $\hat{x}^{1,\lambda}$ above PT? The τ 's which can be associated to λ obey

$$0 < \text{EqDR}(\nu, \tau) \leq \delta,$$

where $\text{EqDR}(\nu, \tau) = \text{EqDR}(\tau; \delta, \rho, \nu, \sigma)$ is the equilibrium detection rate for a signal with distribution ν . Equivalently, they are those τ where the equilibrium discovery number is n or smaller.

Lemma 6.4.6. *For each $\tau > 0$, obeying both*

$$\text{mse}(0, \tau) < \delta \quad \text{and} \quad \text{EqDR}(\nu, \tau) < \delta,$$

the parameter $\lambda \geq 0$ defined by the calibration relation

$$\lambda(\tau) = \frac{\tau}{\sqrt{1-\gamma}} \cdot (1 - \text{EqDR}(\nu, \tau)/\delta),$$

has the formal MSE

$$\text{fMSE}(\nu, \tau | \text{LASSO}) = \frac{\delta\gamma}{1-\gamma}.$$

One can check that, for each $\lambda \geq 0$, for each phase space point above phase transition, the above construction allows to construct a measure μ with $\epsilon = \delta\rho$ mass on nonzeros and with arbitrarily high formal MSE. This completes the derivation of part 2 of Proposition 6.3.7.

6.5 Empirical validation

So far our discussion explains how state evolution calculations are carried out so others might reproduce them. The actual ‘science contribution’ of this chapter comes in showing that these calculations describe the actual behavior of solutions to (6.2). We check these calculations in two ways: first, to show that individual MSE predictions are accurate, and second, to show that the mathematical structures (least-favorable, minimax saddlepoint, maximin threshold) that lead to our predictions are visible in empirical results.

6.5.1 Below phase transition

Let $\text{fMSE}(\lambda; \delta, \rho, \sigma, \nu)$ denote the formal MSE we assign to $\hat{x}^{1,\lambda}$ for problem instances from $\text{LSF}(\delta, \rho, \sigma, \nu)$. Let $\text{eMSE}(\lambda)_{n,N}$ denote the empirical MSE of the LASSO estimator $\hat{x}^{1,\lambda}$ in a problem instance drawn from $\text{LSF}(\delta, \rho, \sigma, \nu)$ at a given problem size n, N . In claiming that the noise sensitivity of $\hat{x}^{1,\lambda}$ is bounded above by $M^*(\delta, \rho)$, we are saying that in empirical trials, the ratio eMSE/σ^2 will not be larger than M^* with statistical significance. We now present empirical evidence for this claim.

δ	ρ	ϵ	$M^\pm(\epsilon)$	$\tau^\pm(\epsilon)$	$\mu^\pm(\epsilon, 0.02)$	$M^*(\delta, \rho)$	$\mu^*(\delta, \rho, 0.02)$	$\tau^*(\delta, \rho)$	λ^*
0.10	0.09	0.01	0.06	1.96	3.74	0.14	5.79	1.96	1.28
0.10	0.14	0.01	0.08	1.83	3.63	0.41	8.24	1.83	0.83
0.10	0.17	0.02	0.09	1.77	3.58	1.20	12.90	1.77	0.51
0.10	0.18	0.02	0.10	1.75	3.57	2.53	18.28	1.75	0.41
0.25	0.13	0.03	0.15	1.54	3.41	0.39	5.46	1.54	0.98
0.25	0.20	0.05	0.20	1.40	3.29	1.12	7.68	1.40	0.62
0.25	0.24	0.06	0.23	1.33	3.24	3.28	12.22	1.33	0.39
0.25	0.25	0.06	0.24	1.31	3.23	6.89	17.31	1.31	0.30
0.50	0.19	0.10	0.32	1.15	3.11	0.90	5.19	1.15	0.70
0.50	0.29	0.14	0.42	1.00	2.99	2.55	7.35	1.00	0.42
0.50	0.35	0.17	0.47	0.92	2.93	7.51	11.75	0.92	0.26
0.50	0.37	0.18	0.48	0.90	2.91	15.75	16.67	0.90	0.20

Table 6.2: Parameters of quasi-least-favorable settings studied in the empirical results presented here.

Accuracy of MSE at the LF signal

We first consider the accuracy of theoretical predictions at the nearly-least-favorable signals generated by $\nu_{\delta, \rho, \alpha} = (1 - \epsilon)\delta_0 + (\epsilon/2)\delta_{-\mu^*(\delta, \rho, \alpha)} + (\epsilon/2)\delta_{\mu^*(\delta, \rho, \alpha)}$ defined by Part 2.b of Proposition 6.3.7. If the empirical ratio eMSE/σ^2 is substantially above the theoretical bound $M^*(\delta, \rho)$, according to standards of statistical significance, we have falsified the proposition.

We consider parameter points $\delta \in \{0.10, 0.25, 0.50\}$ and $\rho \in \{\frac{1}{2} \cdot \rho_{\text{MSE}}, \frac{3}{4} \cdot \rho_{\text{MSE}}, \frac{9}{10} \cdot \rho_{\text{MSE}}, \frac{19}{20} \cdot \rho_{\text{MSE}}\}$. The predictions of the SE formalism are detailed in Table 6.2.

Results at $N = 1500$

To test these predictions, we generate in each situation $R = 200$ random realizations of size $N = 1500$ from $\text{LSF}(\delta, \rho, \sigma, \nu)$ with the parameters shown in Table 6.2 and run the LARS/LASSO solver to find the solution $\hat{x}^{1, \lambda}$. Table 6.3 shows the empirical average MSE in 200 trials at each tested situation.

Except at $\delta = 0.10$ the mismatch between empirical and theoretical a few to several percent - reasonable given the sample size $R = 200$. At $\delta = 0.10$, $\rho = 0.180$

δ	ρ	μ	λ^*	fMSE	eMSE	SE
0.100	0.095	5.791	1.258	0.136	0.126	0.0029
0.100	0.142	8.242	0.804	0.380	0.329	0.0106
0.100	0.170	12.901	0.465	1.045	0.755	0.0328
0.100	0.180	18.278	0.338	2.063	1.263	0.0860
0.250	0.134	5.459	0.961	0.374	0.373	0.0046
0.250	0.201	7.683	0.592	1.028	1.002	0.0170
0.250	0.241	12.219	0.351	2.830	2.927	0.0733
0.250	0.254	17.314	0.244	5.576	5.169	0.1978
0.500	0.193	5.194	0.689	0.853	0.836	0.0078
0.500	0.289	7.354	0.400	2.329	2.251	0.0254
0.500	0.347	11.746	0.231	6.365	6.403	0.1157
0.500	0.366	16.667	0.159	12.427	11.580	0.2999

Table 6.3: Results at $N = 1500$. MSE of $\text{LASSO}(\lambda^*)$ at nearly-least-favorable situations, together with standard errors (SE)

– close to phase transition – there is a mismatch needing attention. (In fact, at each level of δ the most serious mismatch is at the value of ρ closest to phase transition. This can be attributed partially to the blowup of the quantity being measured as we approach phase transition.) We will pursue this mismatch below.

We also ran trials at $\delta \in \{0.15, 0.20, 0.30, 0.35, 0.40, 0.45\}$. These cases exhibited the same patterns seen above, with adequate fit except at small δ , especially near phase transition. We omit the data here.

In all our trials, we measured numerous observables – not only the MSE. The trend in mismatch between theory and observation in such observables was comparable to that seen for MSE.

Results at $N = 4000$

Statistics of random sampling dictate that there always be some measure of disagreement between empirical averages and expectations. When the expectations are taken in the large-system limit, as ours are, there are additional small- N effects that appear separate from random sampling effects. However, both sorts of effects should visibly decline with increasing N .

δ	ρ	μ	λ^*	fMSE	eMSE	SE
0.100	0.095	5.791	1.258	0.136	0.128	0.0016
0.100	0.142	8.242	0.804	0.380	0.348	0.0064
0.100	0.170	12.901	0.465	1.045	0.950	0.0228
0.100	0.180	18.278	0.338	2.063	1.588	0.0619
0.250	0.134	5.459	0.961	0.374	.371	0.0028
0.250	0.201	7.683	0.592	1.028	1.023	0.0106
0.250	0.241	12.219	0.351	2.830	2.703	0.0448
0.250	0.254	17.314	0.244	5.576	5.619	0.0428
0.500	0.193	5.194	0.689	0.853	0.849	0.0047
0.500	0.289	7.354	0.400	2.329	2.296	0.016
0.500	0.347	11.746	0.231	6.365	6.237	0.0677
0.500	0.366	16.667	0.159	12.427	12.394	0.171

Table 6.4: Results at $N = 4000$. Theoretical and empirical MSE's of $\text{LASSO}(\lambda^*)$ at nearly-least-favorable situations, together with standard errors (SE).

Table 6.4 presents results for $N = 4000$; we expect the discrepancies to shrink when the experiments are run at larger value of N . We study the same ρ and δ that were studied for $N = 1500$, and see that the mismatches in our MSE's have grown smaller with N .

Results at $N = 8000$

Small values of δ have the largest discrepancy specially when ρ is chosen very close to the phase transition curve. To show that this discrepancy shrinks as we increase the value of N , we do a similar experiment for $\delta = 0.10$ but this time with $N = 8000$. Table 6.5 summarizes the results of this simulation and shows better agreement between the formal predictions and empirical results.

The alert reader will no doubt have noticed that the discrepancy between theoretical predictions and empirical results is in many cases quite a bit larger in magnitude than the size of the formal standard errors reported in the above tables. We emphasize that the theoretical predictions are formal limits for the $N \rightarrow \infty$ case, while empirical results take place at finite N . In both statistics and statistical physics it

δ	ρ	μ	λ^*	fMSE	eMSE	SE
0.100	0.095	5.791	1.258	0.136	0.131	0.0012
0.100	0.142	8.242	0.804	0.380	0.378	0.0046
0.100	0.170	12.901	0.465	1.045	1.024	0.0186
0.100	0.180	18.278	0.338	2.063	1.883	0.0458

Table 6.5: Results at $N = 8000$. Theoretical and empirical MSE's of LASSO(λ^*) at nearly-least-favorable situations with $\delta = 0.10$, together with standard errors (SE) of the empirical MSE's

is quite common for mismatches between finite- N results and N -large to occur as either $O(N^{-1/2})$ (e.g. Gaussian approximation to the Poisson) or $O(N^{-1})$ effects (e.g. Gaussian approximation to fair coin tossing). Analogously, we might anticipate that mismatches in this setting of order $N^{-\alpha}$ with α either $1/2$ or 1 . Figure 6.9 presents empirical and theoretical results taken from the cases $N = 1500, 4000, \text{ and } 8000$ and displays them on a common graph, with y -axis a mean-squared error (empirical or theoretical) and on the x axis the inverse system size $1/N$. The case $1/N = 0$ presents the formal large-system limit predicted by our calculations and the other cases $1/N > 0$ present empirical results described in the tables above. As can be seen, the discrepancy between formal MSE and empirical MSE tends to zero linearly with $1/N$. (A similar plot with $1/\sqrt{N}$ on the x -axis would not be so convincing.)

Finding 6. *The formal and empirical MSE's at the quasi saddlepoint (ν^*, λ^*) show statistical agreement at the cases studied, in the sense that either the MSE's are consistent with standard statistical sampling formulas, or, where they were not consistent at $N = 1500$, fresh data at $N = 4000$ and $N = 8000$ showed marked reductions in the anomalies confirming that the anomalies decline with increasing N .*

Existence of game-theoretic saddlepoint in eMSE

Underlying our derivations of minimax formal MSE is a game-theoretic saddlepoint structure, illustrated in Figure 6.10. The loss function MSE has the following structure around the quasi saddlepoint (ν^*, λ^*) : any variation of μ to lower values, will cause a reduction in loss, while a variation of λ to other values will cause an increase

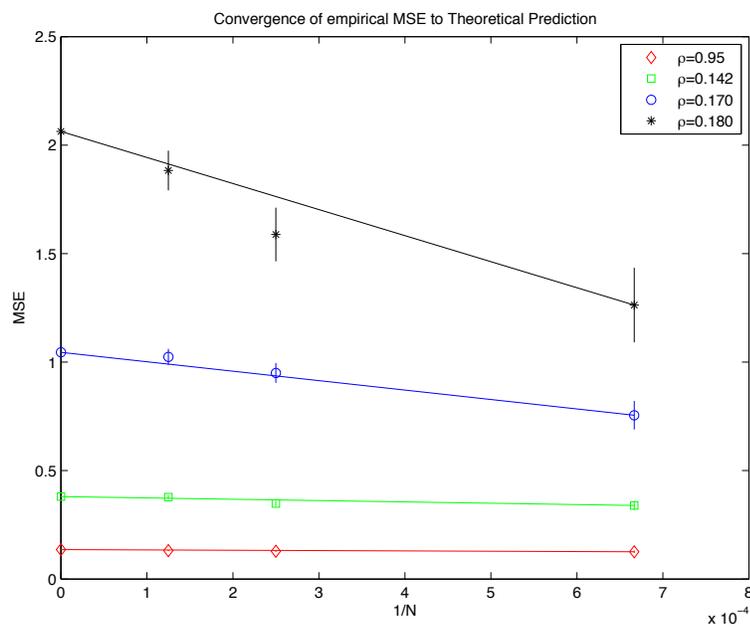


Figure 6.9: Finite- N scaling of empirical MSE. Empirical MSE results from the cases $N = 1500$, $N = 4000$ and $N = 8000$ and $\delta = 0.1$. Vertical axis: empirical MSE. Horizontal axis: $1/N$. Different colors/symbols indicate different values of the sparsity control parameter δ . Vertical bars denote $\pm 2SE$ limits. Theoretical predictions for the $N = \infty$ case appear at $1/N = 0$. Lines connect the cases $N = 1500$ and $N = \infty$.

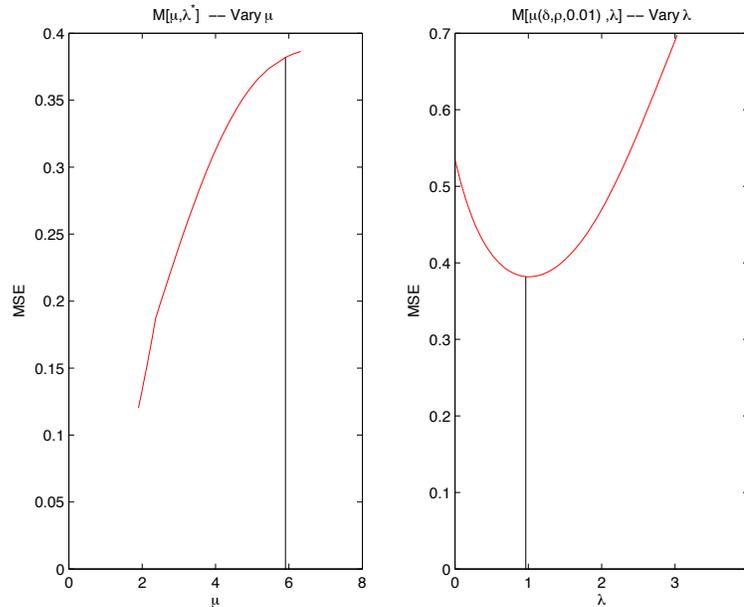


Figure 6.10: Saddlepoint in formal MSE. Right panel: Behavior of formal MSE as λ is varied away from λ^* . Left panel: Behavior of formal MSE as μ is varied away from μ^* in the direction of smaller values. Black lines indicate locations of μ^* and λ^* . $\delta = 0.25$, $\rho = \rho_{\text{MSE}}(\delta)/2$.

in loss.

Other penalization gives larger MSE

If our formalism is correct in deriving optimal penalization for $\hat{x}^{1,\lambda}$, we will see that changes of the penalization away from λ^* will cause MSE to increase. We consider the same situations as earlier, but now vary λ away from the minimax value, while holding the other aspects of the problem fixed. In the appendix, Tables 6.7 and 6.8 presents numerical values of the empirical MSE obtained. Note the agreement of formal MSE, in which a saddlepoint is rigorously proven, and empirical MSE, which represents actual LARS/LASSO reconstructions. Also in this case we used $R = 200$ Monte Carlo replications.

To visualize the information in those tables, we refer to Figure 6.11.

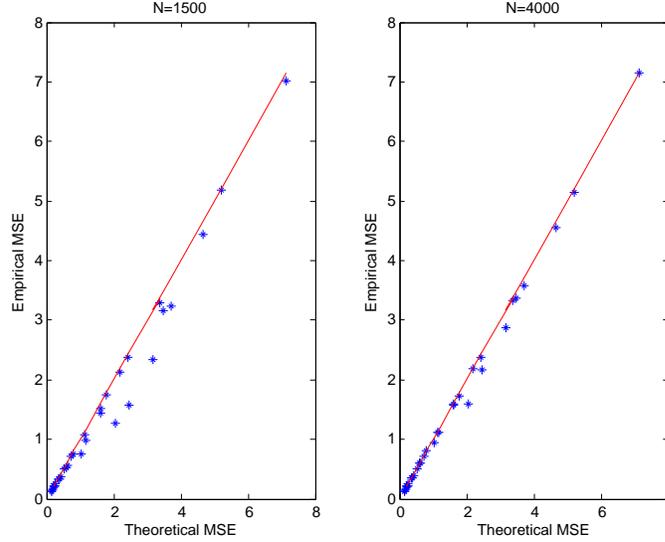


Figure 6.11: Scatterplots comparing Theoretical and Empirical MSE's found in Tables 6.7 and 6.8. Left Panel: results at $N = 1500$. Right Panel: results at $N = 4000$. Note visible tightening of the scatter around the identity line as N increases.

MSE with more favorable measures is smaller

In our formalism, fixing $\lambda = \lambda^*$, and varying μ to smaller values will cause a reduction in formal MSE. Namely, if instead of $\mu^*(\delta, \rho, 0.01)$ we used $\mu^*(\delta, \rho, \alpha)$ for α significantly larger than 0.01, we would see a significant reduction in MSE, by an amount matching the predicted amount.

Recall that $\text{mse}(\nu, \tau)$ denotes the ‘risk’ (MSE) of scalar soft thresholding as in Section 6.2, with input distribution ν , noise variance 1, and threshold τ . Now suppose that $\text{mse}(\nu_0, \tau) > \text{mse}(\nu_1, \tau)$. Then also the resulting formal noise-plus-interference obeys $\text{fNPI}(\nu_0, \tau) > \text{fNPI}(\nu_1, \tau)$. As noticed several times in Section 6.4.4, the formal MSE of AMPT obeys $\text{fMSE}(\nu, \tau) = \text{mse}(\tilde{\nu}, \tau) \cdot \text{fNPI}(\nu, \tau)$, where $\tilde{\nu}$ denotes a rescaled probability measure (as in the proof of Proposition 6.4.5). Hence

$$\text{fMSE}(\nu_1, \tau) \leq \text{mse}(\tilde{\nu}_1, \tau) \cdot \text{fNPI}(\nu_0, \tau),$$

where the scaling uses $\text{fNPI}(\nu_0)$. In particular, for $\mu = \mu^*(\delta, \rho, \alpha) = \mu^\pm(\delta, \rho, \alpha) \sqrt{\text{NPI}^*(\delta, \rho)}$,

the three point mixture: $\nu_{\delta,\rho,\alpha}$ has

$$\text{fMSE}(\nu_{\delta,\rho,\alpha}, \tau^*) \leq (1 - \alpha)M^*(\delta, \rho),$$

and we ought to be able to see this. Table 6.9 shows results of simulations at $N = 1500$. The theoretical MSE drops as we move away from the nearly least favorable μ in the direction of smaller μ , and the empirical MSE responds similarly.

Finding 7. *The empirical data exhibit the saddlepoint structures predicted by the SE formalism.*

MSE of mixtures

The SE formalism contains a basic mathematical structure which allows one to infer that behavior at one saddlepoint determines the global minimax value: behavior under taking convex combinations (mixtures) of measures ν .

Let $\text{mse}(\nu, \lambda)$ denote the ‘risk’ (MSE) of scalar soft thresholding as in Section 6.2. For such scalar thresholding, we have the affine relation

$$\text{mse}((1 - \gamma)\nu_0 + \gamma\nu_1, \tau) = (1 - \gamma)\text{mse}(\nu_0, \tau) + \gamma \cdot \text{mse}(\nu_1, \tau).$$

Now suppose that $\text{mse}(\nu_0, \tau) > \text{mse}(\nu_1, \tau)$. Then also $\text{NPI}(\nu_0, \tau) > \text{NPI}(\nu_1, \tau)$. The formal MSE of AMPT obeys the scaling relation $\text{fMSE}(\nu, \tau) = \text{mse}(\tilde{\nu}, \tau) \cdot \text{NPI}(\nu, \tau)$, where $\tilde{\nu}$ denotes the rescaled probability measure, argument rescaled by $1/\sqrt{NPI}$.

We conclude that

$$\text{fMSE}((1 - \gamma)\nu_0 + \gamma\nu_1, \tau) \leq (1 - \gamma) \cdot \text{mse}(\tilde{\nu}_0, \tau) \cdot \text{NPI}(\nu_0, \tau) + \gamma \cdot \text{mse}(\tilde{\nu}_1, \tau) \cdot \text{NPI}(\nu_0, \tau), \quad (6.25)$$

This ‘quasi-affinity’ relation allows to extend the saddlepoint structure from 3 point mixtures to more general measures.

To check this, we consider two near-least-favorable measures, $\nu_0 = \nu_{\delta,\rho,0.02}$ and $\nu_1 = \nu_{\delta,\rho,0.50}$. and generate a range of cases $\nu^{(\alpha)} = (1 - \alpha)\nu_0 + \alpha\nu_1$ by varying alpha. When $\alpha \notin \{0, 1\}$ this is a 5 point mixture rather than one of the 3-point

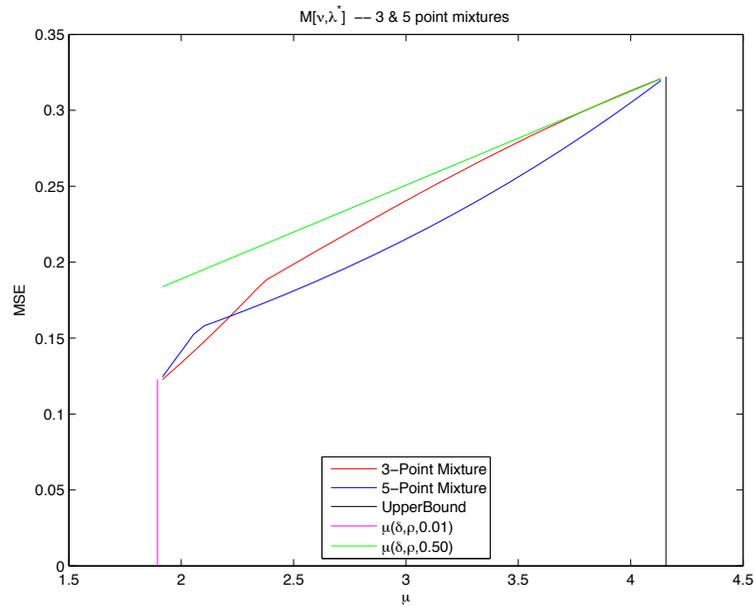


Figure 6.12: Convexity structures in formal MSE. Behavior of formal MSE of 5 point mixture combining nearly least-favorable μ with discount of 1% and one with discount of 50%. Also, the convexity bound (6.25) and the formal MSE of associated 3-point mixtures is displayed. $\delta = 0.25$, $\rho = \rho_{\text{MSE}}(\delta)/2$.

mixtures we have been studying. Figure 6.12 displays the convexity bound (6.25), and the behavior of the formal MSE of this 5 point mixture. For comparison it also presents the formal MSE of the 3 point mixture having its mass at the weighted mean $(1 - \alpha)\mu(\delta, \rho, 0.02) + \alpha\mu(\delta, \rho, 0.50)$. Evidently, the 5 point mixture typically has smaller MSE than the comparable 3-point mixture, and it always is below the convexity bound.

Finding 8. *The empirical MSE obeys the mixture inequalities predicted by the SE formalism.*

6.5.2 Above phase transition

We conducted an empirical study of the formulas derived in Section 6.4.5. At $\delta = 0.25$ we chose $\rho = 0.401$ - well above phase transition - and selected a range of τ and γ values allowed by our formalism. For each pair γ, τ , we generated $R = 200$ Monte Carlo realizations and obtained LASSO solutions with the given penalization parameter λ . The results are described in Table 6.6. The match between formal MSE and empirical MSE is acceptable.

Finding 9. *Running $\hat{x}^{1,\lambda}$ at the 3-point mixtures defined for the regime above phase transition in Lemma 6.4.6 yields empirical MSE consistent with the formulas of that Lemma.*

This validates the unboundedness of MSE of LASSO above phase transition.

6.6 Extensions

6.6.1 Positivity constraints

A completely parallel treatment can be given for the case where $x^0 \geq 0$. In that setting, we use the positivity-constrained soft-threshold

$$\eta^+(x; \theta) = \begin{cases} x - \theta & \text{if } \theta < x, \\ 0 & \text{if } x \leq \theta, \end{cases} \quad (6.26)$$

and consider the corresponding positive-constrained thresholding minimax MSE [37]

$$M^+(\epsilon) = \inf_{\tau > 0} \sup_{\nu \in \mathcal{F}_\epsilon^+} \mathbb{E} \left\{ \left[\eta^+(X + \sigma \cdot Z; \tau\sigma) - X \right]^2 \right\}, \quad (6.27)$$

where

$$\mathcal{F}_\epsilon^+ = \{ \nu : \nu \text{ is probability measure with } \nu[0, \infty) = 1, \nu(\{0\}) \geq 1 - \epsilon \}.$$

We consider the positive-constrained ℓ_1 -penalized least-squares estimator $x^{1,\lambda,+}$, the solution to

$$(P_{2,\lambda,1}^+) \quad \text{minimize}_{x \geq 0} \quad \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1. \quad (6.28)$$

We define the minimax, formal *noise sensitivity*:

$$M^{+,*}(\delta, \rho) = \sup_{\sigma > 0} \max_{\nu} \min_{\lambda} \text{fMSE}(x^{1,\lambda,+}, \nu, \sigma^2) / \sigma^2; \quad (6.29)$$

here $\nu \in \mathcal{F}_{\rho\delta}^+$ is the marginal distribution of x_0 . Let $\rho_{\text{MSE}}^+(\delta)$ denote the solution of

$$M^+(\rho\delta) = \delta. \quad (6.30)$$

In complete analogy to (6.5) we have the formula:

$$M^{+,*}(\delta, \rho) = \begin{cases} \frac{M^+(\delta\rho)}{1 - M^+(\delta\rho)/\delta}, & \rho < \rho_{\text{MSE}}^+(\delta), \\ \infty, & \rho \geq \rho_{\text{MSE}}^+(\delta). \end{cases} \quad (6.31)$$

The argument is the same as above, using the AMP formalism, with obvious modifications. All other features of Proposition 6.3.7 carry over, with obvious substitutions. Figure 6.13 shows the phase transition for the positivity constrained case, as well as the contour lines of $M^{+,*}$. Again in analogy to the sign-unconstrained case, the phase boundary ρ_{MSE}^+ occurs at precisely the same location at the phase boundary for ℓ_1 - ℓ_0 equivalence;

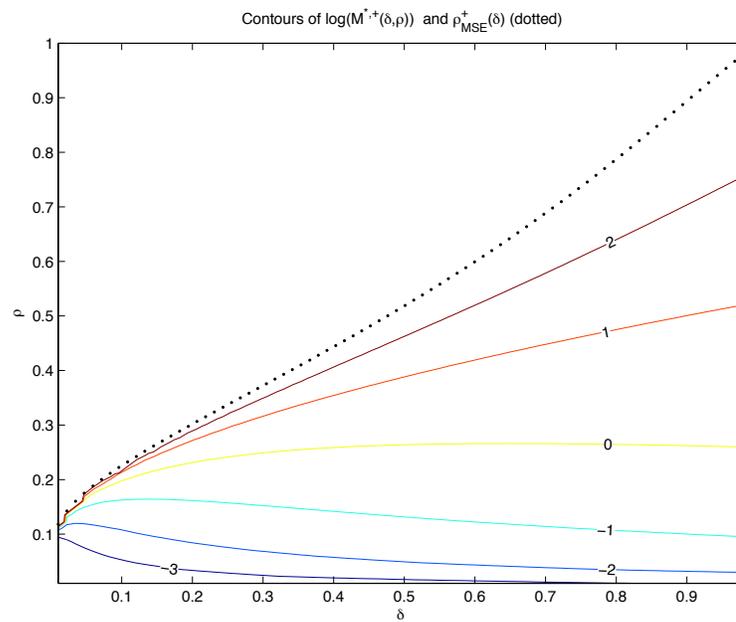


Figure 6.13: Contour lines of the positivity-constrained minimax noise sensitivity $M^{*,+}(\delta, \rho)$ in the (ρ, δ) plane. The dotted black curve graphs the phase boundary $(\delta, \rho_{MSE}^+(\delta))$. Above this curve, $M^{*,+}(\delta, \rho) = \infty$. The colored lines present level sets of $M^{*,+}(\delta, \rho) = 1/8, 1/4, 1/2, 1, 2, 4$ (from bottom to top).

6.6.2 Other classes of matrices

We focused here on matrices A with Gaussian iid entries.

Previously, extensive empirical evidence was presented by Donoho and Tanner [46], that pure ℓ_1 -minimization has its ℓ_1 - ℓ_0 equivalence phase transition at the boundary ρ_{MSE}^\pm not only for Gaussian matrices but for a wide collection of ensembles, including partial Fourier, partial Hadamard, expander graphs, iid ± 1 . This is the noiseless, $\lambda = 0$ case of the general noisy, $\lambda \geq 0$ case studied here.

We believe that similar results to those obtained here hold for matrices A with uniformly bounded iid entries with zero mean and variance $1/n$. In fact, we believe our results should extend to a broader universality class including matrices with iid entries with same mean and variance, under an appropriate light tail condition.

6.7 Tables

This appendix contains table of empirical results supporting our claims.

δ	ρ	γ	μ	τ	λ	fMSE	eMSE
0.250	0.401	0.75	2.8740	1.500	0.9840	0.750	0.746
0.250	0.401	0.85	4.142	1.500	1.168	1.417	1.425
0.250	0.401	0.90	5.345	1.500	1.366	2.250	2.239
0.250	0.401	0.95	7.954	1.500	1.841	4.750	4.724
0.250	0.401	0.97	10.4781	1.500	2.328	8.083	8.126
0.250	0.401	0.98	12.9628	1.500	2.822	12.250	12.327
0.250	0.401	0.99	18.5172	1.500	3.949	24.750	24.601
0.250	0.401	0.995	26.3191	1.500	5.5558	49.750	49.837
0.250	0.401	0.75	2.9031	2.000	2.8766	1.417	1.409
0.250	0.401	0.85	4.058	2.000	3.626	2.250	2.238
0.250	0.401	0.90	5.158	2.000	4.385	2.250	2.238
0.250	0.401	0.95	7.560	2.000	6.122	4.750	4.742
0.250	0.401	0.97	9.897	2.000	7.861	8.083	8.054
0.250	0.401	0.98	12.205	2.000	9.6019	12.250	12.215
0.250	0.401	0.99	17.380	2.000	13.5425	24.750	24.634
0.250	0.401	0.995	24.662	2.000	19.1260	49.750	49.424
0.250	0.401	0.75	2.817	2.500	4.501	1.417	1.409
0.250	0.401	0.85	3.896	2.500	5.750	2.250	2.241
0.250	0.401	0.90	4.926	2.500	7.004	2.250	2.241
0.250	0.401	0.95	7.181	2.500	9.848	4.750	4.712
0.250	0.401	0.97	9.380	2.500	12.6846	8.083	8.050
0.250	0.401	0.98	11.555	2.500	15.5170	12.250	12.215
0.250	0.401	0.99	16.436	2.500	21.9183	24.750	24.619
0.250	0.401	0.995	23.311	2.500	30.9786	49.750	49.442
0.250	0.401	0.75	2.7649	3.000	5.8144	1.417	1.408
0.250	0.401	0.85	3.809	3.000	7.4730	2.250	2.241
0.250	0.401	0.90	4.806	3.000	9.131	2.250	2.241
0.250	0.401	0.95	6.991	3.000	12.880	4.750	4.735
0.250	0.401	0.97	9.125	3.000	16.6113	8.083	8.053
0.250	0.401	0.98	11.236	3.000	20.3339	12.250	12.218
0.250	0.401	0.99	15.975	3.000	28.7413	24.750	24.621
0.250	0.401	0.995	22.652	3.000	40.6356	49.750	49.419

Table 6.6: Results above Phase transition. Parameters of the construction as well as theoretical predictions and resulting empirical MSE figures

Table 6.7: $N = 1500$, λ dependence of the MSE at fixed μ

δ	ρ	μ	λ	fMSE	eMSE	SE
0.100	0.095	5.791	0.402	0.152	0.140	0.0029
0.100	0.095	5.791	1.258	0.136	0.126	0.0029
0.100	0.095	5.791	3.169	0.174	0.164	0.0028
0.100	0.095	5.791	4.948	0.239	0.228	0.0025
0.100	0.142	8.242	0.804	0.380	0.329	0.0106
0.100	0.142	8.242	1.960	0.408	0.374	0.0087
0.100	0.142	8.242	3.824	0.534	0.504	0.0084
0.100	0.142	8.242	6.865	0.737	0.716	0.0059
0.100	0.180	18.278	0.338	2.063	1.263	0.0860
0.100	0.180	18.278	2.934	2.467	1.573	0.0741
0.100	0.180	18.278	7.545	3.474	3.167	0.0569
0.100	0.180	18.278	14.997	4.677	4.438	0.0321
0.250	0.134	5.459	0.518	0.403	0.390	0.0044
0.250	0.134	5.459	0.961	0.374	0.373	0.0046
0.250	0.134	5.459	2.165	0.452	0.455	0.0053
0.250	0.134	5.459	3.555	0.623	0.612	0.0042
0.250	0.201	7.683	0.036	1.151	1.155	0.0174
0.250	0.201	7.683	0.592	1.028	1.002	0.0170
0.250	0.201	7.683	2.243	1.324	1.293	0.0158
0.250	0.201	7.683	4.392	1.861	1.837	0.0114
0.250	0.254	17.314	0.244	5.576	5.169	0.1978
0.250	0.254	17.314	1.433	6.291	5.992	0.1712
0.250	0.254	17.314	3.855	8.667	8.492	0.1148
0.250	0.254	17.314	8.886	12.154	11.978	0.0697
0.500	0.193	5.194	0.176	1.121	1.108	0.0080
0.500	0.193	5.194	0.470	0.894	0.879	0.0070
0.500	0.193	5.194	0.933	0.866	0.862	0.008
0.500	0.193	5.194	1.355	0.965	0.960	0.0078
0.500	0.193	5.194	2.237	1.273	1.263	0.0075
0.500	0.289	7.354	0.179	2.489	2.438	0.0262
0.500	0.289	7.354	0.400	2.329	2.251	0.0254
0.500	0.289	7.354	0.655	2.377	2.329	0.0268
0.500	0.289	7.354	1.137	2.728	2.718	0.0256
0.500	0.289	7.354	2.258	3.704	3.672	0.0212
0.500	0.366	16.666	0.159	12.427	11.580	0.2998
0.500	0.366	16.666	0.582	13.300	13.565	0.2851
0.500	0.366	16.666	1.491	17.028	17.194	0.2082
0.500	0.366	16.666	3.769	23.994	23.571	0.1409

Table 6.8: $N = 4000$, λ dependence of the MSE at fixed μ

δ	ρ	μ	λ	fMSE	eMSE	SE
0.100	0.095	5.791	0.402	0.152	0.144	0.0017
0.100	0.095	5.791	1.258	0.136	0.128	0.0016
0.100	0.095	5.791	2.037	0.142	0.133	0.0016
0.100	0.095	5.791	3.169	0.174	0.168	0.0016
0.100	0.095	5.791	4.948	0.239	0.228	0.0012
0.100	0.142	8.242	0.804	0.380	0.348	0.0064
0.100	0.142	8.242	1.960	0.408	0.389	0.0058
0.100	0.142	8.242	3.824	0.534	0.510	0.0051
0.100	0.142	8.242	6.865	0.737	0.716	0.0034
0.100	0.180	18.278	0.338	2.063	1.588	0.0619
0.100	0.180	18.278	2.934	2.467	2.171	0.0532
0.100	0.180	18.278	7.545	3.474	3.367	0.0312
0.100	0.180	18.278	14.997	4.677	4.551	0.0169
0.150	0.109	5.631	0.420	0.236	0.228	0.0022
0.150	0.109	5.631	1.073	0.212	0.209	0.0023
0.150	0.109	5.631	1.700	0.218	0.213	0.0021
0.150	0.109	5.631	2.657	0.260	0.251	0.0024
0.150	0.109	5.631	4.284	0.359	0.353	0.0017
0.150	0.163	8.030	0.720	0.588	0.595	0.0072
0.150	0.163	8.030	1.614	0.626	0.610	0.0078
0.150	0.163	8.030	3.135	0.804	0.807	0.0058
0.150	0.163	8.030	5.868	1.125	1.118	0.0047
0.150	0.207	17.814	0.305	3.185	2.864	0.0861
0.150	0.207	17.814	2.231	3.715	3.582	0.0722
0.150	0.207	17.814	5.879	5.202	5.141	0.0439
0.150	0.207	17.814	12.455	7.142	7.154	0.0269

Table 6.9: $N = 1500$, μ dependence of the MSE at fixed λ

δ	ρ	μ	λ	fMSE	eMSE	SE
0.100	0.095	5.291	1.253	0.131	0.125	0.0022
0.100	0.095	5.541	1.256	0.134	0.132	0.0025
0.100	0.095	5.691	1.257	0.135	0.126	0.0027
0.100	0.095	5.791	1.258	0.136	0.129	0.0024
0.100	0.095	5.891	1.259	0.137	0.125	0.0027
0.100	0.095	6.041	1.260	0.138	0.126	0.0030
0.100	0.095	6.291	1.262	0.139	0.127	0.0028
0.100	0.095	6.791	1.264	0.141	0.125	0.0031
0.100	0.142	7.242	0.794	0.349	0.317	0.0074
0.100	0.142	7.742	0.800	0.366	0.335	0.0084
0.100	0.142	7.992	0.802	0.373	0.351	0.0089
0.100	0.142	8.000	0.802	0.373	0.362	0.0094
0.250	0.134	4.459	0.952	0.338	0.336	0.0036
0.250	0.134	5.209	0.959	0.367	0.356	0.0044
0.250	0.134	5.459	0.961	0.374	0.362	0.0047
0.250	0.134	5.559	0.962	0.376	0.367	0.0045
0.250	0.134	5.709	0.962	0.379	0.372	0.0048
0.250	0.134	5.959	0.963	0.383	0.362	0.0052
0.250	0.134	6.459	0.964	0.387	0.387	0.0058
0.250	0.201	6.683	0.587	0.939	0.899	0.0126
0.250	0.201	7.183	0.590	0.988	0.965	0.0147
0.250	0.201	7.433	0.591	1.009	0.956	0.0147
0.250	0.201	7.583	0.592	1.021	1.027	0.0155
0.500	0.193	4.194	0.684	0.769	0.770	0.0052
0.500	0.193	4.694	0.687	0.818	0.823	0.0066
0.500	0.193	4.944	0.688	0.837	0.838	0.0073
0.500	0.193	5.294	0.689	0.858	0.845	0.0079
0.500	0.193	5.444	0.690	0.865	0.863	0.0079
0.500	0.193	5.694	0.690	0.874	0.887	0.0085
0.500	0.193	6.194	0.691	0.886	0.868	0.0085
0.500	0.289	6.354	0.398	2.119	2.071	0.0195
0.500	0.289	6.854	0.399	2.234	2.214	0.0235
0.500	0.289	7.254	0.400	2.313	2.271	0.0244
0.500	0.289	7.454	0.400	2.346	2.287	0.0287
0.500	0.289	7.604	0.400	2.370	2.327	0.0306
0.500	0.289	7.854	0.401	2.404	2.339	0.0284
0.500	0.289	8.000	0.401	2.422	2.409	0.0300

Table 6.10: $N = 1500$, MSE for 5-point prior

δ	ρ	μ	λ	Theoretical MSE	Empirical MSE	α
0.250	0.134	1.894	0.857	0.120	0.151	0
0.250	0.134	2.171	0.897	0.162	0.163	0.122
0.250	0.134	2.447	0.901	0.178	0.177	0.244
0.250	0.134	2.724	0.906	0.196	0.195	0.366
0.250	0.134	3.001	0.912	0.215	0.210	0.488
0.250	0.134	3.277	0.918	0.237	0.236	0.611
0.250	0.134	3.554	0.926	0.261	0.257	0.7333
0.250	0.134	3.830	0.935	0.287	0.280	0.8556
0.250	0.134	4.107	0.945	0.317	0.307	0.9778
0.250	0.134	4.383	0.957	0.348	0.359	1.1000

Chapter 7

AMP Versus Other First Order Methods

Consider the simplest form of a convex optimization problem

$$\min_{x \in \mathbb{R}^N} f(x), \tag{7.1}$$

where f is a differentiable convex function. Iterative approaches are used to approximate the optimal point when the exact solution cannot be calculated explicitly. According to the type of information an iterative algorithm uses at each step, it may be categorized as zeroth, first, second, ..., order method. Suppose that the algorithm is iterative and our estimate of the optimal point x^* at iteration t is called x^t . Zeroth-order methods just use the values of $f(x^1), f(x^2), \dots, f(x^t)$ to propose the next estimate. First-order methods however are allowed to use the gradients of the function at those points as well. Clearly, second-order methods also exploit the Hessian of the function and so on. From the computational complexity point of view, each iteration of zeroth-order method is using cheaper information than the first-order method, and the computational complexity of each iteration increases as the order goes up. However, it should be mentioned that the higher order methods are able to make a ‘bigger step’ toward the solution at each step. An example of a zeroth-order method is to first find a compact set in which the optimum lies and define a grid

on that set and evaluate the function at all the points of the grid. An example of a first-order method is gradient descent; finally, Newton's method is a well-known second-order method. Following [83] we define the function class $C_L^{k,\ell}(Q)$: The class of convex functions obeying

1. f is k times continuously differentiable.
2. Its ℓ^{th} derivative is Lipschitz Continuous on Q with Lipschitz constant L .

Also we will use the following definition several times in this chapter. Let $(H, \|\cdot\|)$ be a Hilbert space.

Definition 7.0.1. *An operator $T : H \rightarrow H$ is nonexpansive if it satisfies the following property:*

$$\|Tx - Ty\| \leq \|x - y\|, \quad \forall (x, y) \in H \times H.$$

7.1 Proximity operator

Our goal in this section is to consider a few different optimization problems and propose seemingly different first-order approaches for solving them. Then we introduce the proximity operator and show how it unifies all these problems. The material of this section is taken from [83], [17], [107], [26] and the interested reader may consult these references for further information.

7.1.1 Unconstrained problem

Let f be a differentiable, convex function and suppose that the goal is to find

$$\min_{x \in \mathbb{R}^N} f(x). \tag{7.2}$$

The most well-known first-order method for the above problem is the gradient descent which uses the following iteration:

$$x^{t+1} = x^t - \alpha^t \nabla f(x^t), \tag{7.3}$$

where α^t is the step size that can be set from many different heuristics including backtracking or Barzilai-Borwein. We will discuss the convergence rate of this algorithm and several ways to improve it later in this chapter.

7.1.2 Optimization over ‘easy’ convex sets

Consider the following optimization problem:

$$\min_{x \in \mathcal{C}} f(x), \quad (7.4)$$

where \mathcal{C} is a closed, convex set. Also suppose that finding the solution of the following minimization is cheap:

$$P_{\mathcal{C}}(x) = \arg \min_{y \in \mathcal{C}} \|y - x\|_2. \quad (7.5)$$

For example, if $\mathcal{C} = \{y : \|y\|_2 \leq 1\}$, $P_{\mathcal{C}}(x) = \frac{x}{\|x\|_2}$, and if $\mathcal{C} = \{y : \|y\|_{\infty} \leq 1\}$, then

$$(P_{\mathcal{C}}(x))_i = \begin{cases} 1 & \text{if } 1 < x_i, \\ x_i & \text{if } -1 \leq x_i \leq 1, \\ -1 & \text{if } x_i < -1. \end{cases}$$

Lemma 7.1.1. *The projection operator $P_{\mathcal{C}}$ is **nonexpansive**:*

$$\|P_{\mathcal{C}}(x) - P_{\mathcal{C}}(y)\|_2 \leq \|x - y\|_2, \quad \forall x, y \in \text{dom}(P_{\mathcal{C}}).$$

$\text{dom}(P_{\mathcal{C}})$ is the domain of $P_{\mathcal{C}}$.

The proof of this statement is simple and therefore we skip it. The following *generalized gradient descent* iteration can be used for solving this problem:

$$x^{t+1} = P_{\mathcal{C}}(x^t - \alpha^t \nabla f(x^t)). \quad (7.6)$$

We will discuss the convergence of this algorithm later in this chapter.

7.1.3 ‘Easy’ non-differentiable functions

Now consider the following optimization problem:

$$\min_{x \in \mathcal{R}^N} f(x) + g(x), \quad (7.7)$$

where, as before, f is a differentiable convex function. g is also convex but non-differentiable. In addition to the above assumptions, suppose that g has the property that it is easy to solve the following optimization problem:

$$\text{Prox}_{\alpha g}(x) = \arg \min \frac{1}{2} \|y - x\|_2^2 + \alpha g(y). \quad (7.8)$$

This is called the proximity operator of the function g . For example a well-known example of the proximity operator is $\text{Prox}_{\alpha g}(x) = \eta(x; \alpha)$ which is the proximity operator of $g(x) = \|x\|_1$. The following two lemmas taken from [26] will be useful in our discussions in this chapter.

Lemma 7.1.2. *Given the proximity operator of a function g we can calculate the proximity operator of its dual function g^* from,*

$$\text{Prox}_{\alpha g^*}(x) = x - \alpha \text{Prox}_{\frac{1}{\alpha} g}\left(\frac{x}{\alpha}\right);$$

Proof. According to the definition,

$$\text{Prox}_{\alpha g^*}(x) = \arg \min_y \frac{1}{2} \|x - y\|_2^2 + \alpha g^*(y).$$

If we write the definition of g^* we obtain

$$\begin{aligned} \min_y \frac{1}{2} \|x - y\|_2^2 + \alpha g^*(y) &= \min_y \frac{1}{2} \|x - y\|_2^2 + \alpha \max_z \langle z, y \rangle - g(y) \stackrel{1}{=} \\ \max_z \min_y \frac{1}{2} \|x - y\|_2^2 + \alpha \langle z, y \rangle - \alpha g(y) &= \max_z -\frac{1}{2} \|\alpha z\|_2^2 + \alpha \langle x, z \rangle - \alpha g(z) = \\ \frac{1}{2} \|x\|_2^2 - \min_z \|x - \alpha z\|_2^2 + \alpha g(z), \end{aligned}$$

where equality 1 is true since the function is convex in terms of y and concave in

terms of z . Therefore, $z^* = \text{Prox}_{\frac{1}{\alpha}g}(\frac{x}{\alpha})$ and $\text{Prox}_{\alpha g^*}(x) = y^* = x - \alpha z^* = x - \alpha \text{Prox}_{\frac{1}{\alpha}g}(\frac{x}{\alpha})$. \square

It is easy to see from the above lemma that $\text{Prox}_g(x) + \text{Prox}_{g^*}(x) = x$; this is called *Moreau's decomposition*. We return to this point in the next section.

Lemma 7.1.3. *For a convex function g , $\text{Prox}_{\alpha g}(x)$ is a non-expansive operator.*

Proof. Suppose that $\partial g(x)$ represents the set of all subgradients of function f at x . For $u \in \partial g(\text{Prox}_{\alpha g}(x))$ we have

$$g(\text{Prox}_{\alpha g}(z)) \geq g(\text{Prox}_{\alpha g}(x)) + \langle u, \text{Prox}_{\alpha g}(z) - \text{Prox}_{\alpha g}(x) \rangle.$$

From the Karush-Kuhn-Tucker or KKT condition we have, $(\text{Prox}_{\alpha g}(x) - x)/\alpha \in \partial g(\text{Prox}_{\alpha g}(x))$. Therefore,

$$g(\text{Prox}_{\alpha g}(z)) \geq g(\text{Prox}_{\alpha g}(x)) + \langle \text{Prox}_{\alpha g}(x) - x, \text{Prox}_{\alpha g}(z) - \text{Prox}_{\alpha g}(x) \rangle.$$

Similarly,

$$g(\text{Prox}_{\alpha g}(x)) \geq g(\text{Prox}_{\alpha g}(z)) + \langle \text{Prox}_{\alpha g}(z) - z, \text{Prox}_{\alpha g}(x) - \text{Prox}_{\alpha g}(z) \rangle.$$

Adding these two we obtain

$$\begin{aligned} \|\text{Prox}_{\alpha g}(x) - \text{Prox}_{\alpha g}(z)\|^2 &\leq \langle \text{Prox}_{\alpha g}(x) - \text{Prox}_{\alpha g}(z), x - z \rangle \\ &\leq \|\text{Prox}_{\alpha g}(x) - \text{Prox}_{\alpha g}(z)\| \|x - z\|. \end{aligned}$$

\square

If we assume that the proximity operator can be calculated easily for function g in (7.7), then we can use generalized gradient descent algorithm for this problem; its main iteration is given by

$$x^{t+1} = \text{Prox}_{\alpha g}(x^t - \alpha^t \nabla f(x^t)). \quad (7.9)$$

We will analyze the convergence rates of these algorithms in Section 7.2.

7.1.4 One framework for all

So far we have mentioned three seemingly different problems and proposed similar approaches for each of them. It turns out that all these algorithms can be derived from a general framework. This framework has already been introduced in the last section. Proximity operators provide us a tool to derive generalized gradient descent algorithm. As a first step to demonstrate this claim let us show that projection operators are proximity operators themselves. Consider a new type of indicator function

$$\iota_{\mathcal{C}}(x) = \begin{cases} 0 & \text{if } x \in \mathcal{C}, \\ \infty & \text{if } x \notin \mathcal{C}. \end{cases}$$

By using this function we can write optimization (7.4) as

$$\min_{x \in \mathbb{R}^N} f(x) + \iota_{\mathcal{C}}(x),$$

which is in the form (7.7). It is not difficult to check that the proximity operator of $\iota_{\mathcal{C}}(x)$ is the projection operator. Now consider the minimization problem (7.7) and suppose that the current estimate of the optimal solution x^* is x^t and the goal is to give a new estimate for time $t + 1$. Since f is differentiable we approximate it with a quadratic function and then do the minimization on the quadratic function

$$x^{t+1} = \arg \min_{z \in \mathbb{R}^N} f(x^t) + \langle \nabla f(x^t), z - x^t \rangle + \frac{1}{\alpha^t} \|z - x^t\|_2^2 + g(z),$$

which results in

$$x^{t+1} = \text{Prox}_{\alpha^t g}(x^t - \alpha^t \nabla f(x^t)). \quad (7.10)$$

If g was zero, the proximity operator would be equal to the identity and the algorithm would be the same as gradient descent algorithm. Another approach to derive this generalized gradient descent algorithm is the majorization-minimization approach.

Suppose that $f \in C_L^{1,1}(\mathbb{R})$. Then, we have

$$f(x) + g(x) \leq f(x^t) + \langle x - x^t, \nabla f(x^t) \rangle + \frac{1}{2\alpha^t} \|x - x^t\|_2^2 + g(x), \quad (7.11)$$

where $\alpha^t < \frac{1}{L}$ is assumed here. In other words, the convex function is majorized by a quadratic cost function plus $g(x)$. Now if we minimize the right hand side, we again obtain the gradient descent algorithm.

Theorem 7.1.4. [107] *Suppose that $f \in C_L^{1,1}(\mathbb{R})$ and $\alpha^t < \frac{1}{L}$. Under these two assumptions, every step of the generalized gradient descent reduces the composite cost function, i.e., $f(x^{t+1}) + g(x^{t+1}) \leq f(x^t) + g(x^t)$. Furthermore, if $f(x^{t+1}) + g(x^{t+1}) = f(x^t) + g(x^t)$ then $x^{t+1} = x^t$ and x^t is one of the minimizers of $f(x) + g(x)$.*

Proof. we know that

$$f(x) + g(x) \leq f(x^t) + \langle x - x^t, \nabla f(x^t) \rangle + \frac{1}{2\alpha^t} \|x - x^t\|_2^2 + g(x), \quad \forall x \in \mathbb{R}^N.$$

Therefore,

$$\begin{aligned} f(x^{t+1}) + g(x^{t+1}) &\leq f(x^t) + \langle x^{t+1} - x^t, \nabla f(x^t) \rangle + \frac{1}{2\alpha^t} \|x^{t+1} - x^t\|_2^2 + g(x^{t+1}) \\ &\stackrel{1}{\leq} f(x^t) + g(x^t). \end{aligned}$$

Inequality (1) is due to the fact that x^{t+1} minimizes the right hand side and hence the proof of the first part is complete. For proving the second part we use the above inequalities. If $f(x^{t+1}) + g(x^{t+1}) = f(x^t) + g(x^t)$ then

$$f(x^{t+1}) + g(x^{t+1}) \leq f(x^t) + \langle x^{t+1} - x^t, \nabla f(x^t) \rangle + \frac{1}{2\alpha^t} \|x^{t+1} - x^t\|_2^2 + g(x^{t+1}).$$

and since $\alpha^t < \frac{1}{L}$ we see that $x^{t+1} = x^t$. Also, since we know that x^{t+1} is the minimizer of (7.11) we see that $-\nabla f(x^t) \in \partial g(x^t)$. \square

7.1.5 First order methods for compressed sensing

So far in this chapter we have focused on general convex optimization problems and mentioned how one can design generalized gradient descent for a class of convex optimization problems. Using the proximity operator for solving the ℓ_1 -minimization problem has been an active area of research in the last decade. Here is a list of papers published in the last several years in this area [24], [63], [28], [49], [26], [47],[70], [115], [57], [106], [56], [55], [113], [113], [50], [51], [64], [6], [9], [75], [38], [84], [7], [18], [86], [16]. To remind our notation, the measurement matrix is called $A \in \mathbb{R}^{n \times N}$ which is drawn at random from a distribution. We measure a signal x_o through the matrix A , i.e., $y = Ax_o + w$, where w is the measurement noise. The ultimate goal is to recover x_o from the measurements y . As mentioned in chapter 3, ℓ_1 -minimization provides the highest phase transition among several algorithms that we examined in that chapter. Suppose that we are interested in solving the following optimization problem:

$$\min_x \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1. \quad (7.12)$$

Since calculating the proximity operator for $\|x\|_1$ is straightforward we use the generalized gradient descent algorithm for solving this problem. The resulting algorithm is

$$x^{t+1} = \eta(x^t + \alpha^t A^*(y - Ax^t); \alpha^t \lambda), \quad (7.13)$$

which is the very similar in form to the iterative soft thresholding algorithm that we suggested before. The main difference is that λ does not change with time. The cost per iteration of this algorithms is very low, however the important question that shall be addressed here is the rate of convergence. There are two different approaches to address this problem. Deterministic and statistical. The most popular approach for analyzing algorithms in convex optimization is deterministic, where we first define a class of functions, and derive an upper bound on the convergence rate of the algorithm that holds for all the problems in this subclass. On the other hand in the statistical approach there is some randomness in the problem. For instance, the measurement matrix A is random in any recovery algorithm of compressed sensing. Therefore, here we change the notion of convergence rate to the expected convergence rate, i.e., we

measure the convergence rate of $\frac{1}{N}\mathbb{E}\|x^t - x^*\|^2$. Since we are usually interested in large problem sizes, we may consider $\lim_{N \rightarrow \infty} \frac{1}{N}\mathbb{E}\|x^t - x^*\|^2$. As we will see in the next section, the deterministic approach is of limited use in compressed sensing and a better way to analyze the algorithms is the statistical approach.

7.2 Analyzing the convergence rate

In this section we first discuss the deterministic approach for calculating the convergence rate of algorithms. We then demonstrate the advantage of the statistical approach by a simple example. Finally, we will explain the statistical approach for compressed sensing problems.

7.2.1 Drawback of deterministic approach

Let \mathcal{F} be a class of convex functions. Also, let \mathcal{A} be a class of iterative algorithms for solving the optimization problems $\min_x f(x)$ for $f \in \mathcal{F}$. We use the notation $x_{\mathbf{a},f}^t$ for the estimate of algorithm $\mathbf{a} \in \mathcal{A}$ on the function $f \in \mathcal{F}$ at time t . Further, assume that X_f^* is the set of minimizers of f . Define $d(x_{\mathbf{a},f}^t, X_f^*) = \inf_{x^* \in X_f^*} \|x_{\mathbf{a},f}^t - x^*\|_2$. Inspired by [83], we define the following two minimax errors at iteration t :

$$MSE^t(\mathcal{F}, \mathcal{A}) = \min_{\mathbf{a} \in \mathcal{A}} \max_{f \in \mathcal{F}} d(x_{\mathbf{a},f}^t, X_f^*),$$

$$PE^t(\mathcal{F}, \mathcal{A}) = \min_{\mathbf{a} \in \mathcal{A}} \max_{f \in \mathcal{F}} [f(x_{\mathbf{a},f}^t) - f(x_f^*)],$$

where X_f^* is the set of minimizers of f and $x_f^* \in X_f^*$. We call the first one minimax mean square error at time t and the second one minimax prediction error at time t . In the next section we will provide a lower bound for the minimax errors on compressed sensing problems.

Lower bound for the deterministic convergence rate

Consider the following class of functions:

$$\mathcal{P}_L^{n,N} = \left\{ f(x) = \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1 : A \in \mathbb{R}^{n \times N} \text{ \& } \|A^* A\|_{2 \rightarrow 2} \leq L \right\}.$$

This is the class of functions we are interested in compressed sensing. Also let \mathcal{A}_{fo} be the class of all first-order methods. The following theorem provides a lower bound on the minimax mean square and prediction errors.

Theorem 7.2.1. *Consider the class of functions $\mathcal{P}_L^{n,N}$. For any given L, n, N , and for any $t \leq \frac{n-1}{2}$,*

$$\begin{aligned} PE^t(\mathcal{P}_L^{n,N}, \mathcal{A}_{fo}) &\geq CL \frac{\|x^0 - x_o\|^2}{(t+1)^2}, \\ MSE^t(\mathcal{P}_L^{n,N}, \mathcal{A}_{fo}) &\geq \frac{1}{5} \|x^0 - x_o\|. \end{aligned}$$

Our proof is very similar to the proof given for differentiable functions in [83]. We will cook up a problem instance for which the performance of the first-order methods is lower bounded by the above formulas. For the complete proof, please refer to F.1.

Note: As is clear from the above theorem and explained in the previous section, the convergence rate given for $\|x^t - x^*\|^2$ is very disappointing and we could not use first-order methods if it was true. However, this result is pessimistic since it considers a problem that has a very specific structure.

Considering the class $\mathcal{P}_L^{n,N}$ we can now analyze the convergence rate of iterative soft thresholding for a fixed value of threshold. Since Theorem 7.2.1 rules out the possibility of providing any deterministic rate for the convergence of $\|x^t - x^*\|^2$ we just consider the convergence of $f(x^t) - f(x_o)$. The following theorem due to [6] bounds the prediction error of the iterative soft thresholding algorithm.

Theorem 7.2.2. [6] *For any $f \in \mathcal{P}_L^{n,N}$, the estimates of iterative soft thresholding, given by*

$$x^{t+1} = \eta(x^t + \frac{1}{L} A^*(y - Ax^t); \frac{\lambda}{L}),$$

satisfies

$$f(x^t) - f(x_*) \leq CL \frac{\|x^0 - x_o\|^2}{t+1}.$$

In 1983 Nesterov [82] suggested an algorithm for differentiable functions to achieve the lower bound of the minimax prediction error derived in Theorem 7.2.1. This algorithm has been extended to the ℓ_1 -minimization problem by using proximal operators; see [6],[84]. The algorithm proceeds as follows:

$$\begin{aligned} x^{t+1} &= \eta(z^t + \alpha^t A^*(y - Az^t); \lambda \alpha^t), \\ z^t &= x^t + \frac{t-1}{t+2}(x^t - x^{t-1}). \end{aligned} \quad (7.14)$$

This algorithm is also known as FISTA which stands for fast iterative soft thresholding algorithm. The next theorem due to [6] and [82] proves that this algorithm achieves the lower bound of the minimax prediction error at time t .

Theorem 7.2.3. *For any $f \in \mathcal{P}_L^{n,N}$, FISTA given by (7.14) obeys,*

$$f(x^t) - f(x_*) \leq CL \frac{\|x^0 - x_o\|^2}{(t+1)^2}.$$

The interested reader may refer to [6].

We finish this section by emphasizing that since the deterministic approach does not provide any bound on the convergence of $\|x^t - x^*\|$, it is of a limited use in compressed sensing.

7.2.2 Linear system: motivating example

In this section we consider an example of an iterative approach for solving linear system of equations and show how the statistical convergence rate can be calculated by using asymptotic arguments. We also compare the statistical error at iteration t with the minimax error introduced in the last section.

Example 7.1. Suppose that $A \in \mathbb{R}^{n \times n}$ is a symmetric matrix and the upper triangular elements and the diagonal elements are iid $N(0, \frac{1}{n})$. We define a new

positive definite matrix $B = (I + \frac{1}{2}A)$. The elements of x_* are again iid $F_X(x)$ with the following two properties, $\mathbb{E}_F(X) = 0, E_F(X^2) = 1$. We form a vector $y = Bx_*$ and the goal is to find x_* with the following iteration:

$$x^{t+1} = x^t + \frac{1}{2}(y - Bx^t).$$

The algorithm starts at $x^0 = 0$. The goal is to calculate the expectation of the mean square error at iteration t in the asymptotic setting.

$$x^t - x_o = \frac{1}{2^t}(I - \frac{1}{2}A)^t(x^0 - x_o).$$

Therefore,

$$\mathbb{E}\frac{1}{n}(\|x^t - x_o\|_2^2) = \frac{1}{n2^{2t}}\mathbb{E}Tr(I - \frac{1}{2}A)^{2t},$$

According to Lemma C.0.8 and Theorem C.0.3,

$$\lim_{n \rightarrow \infty} \frac{1}{n2^{2t}}\mathbb{E}Tr(I - \frac{1}{2}A)^{2t} = \frac{1}{2^{2t}}\mathbb{E}(1 - \frac{\gamma}{2})^{2t},$$

where the second expectation is with respect to Wigner semicircle law, explained in Appendix C.

$$\begin{aligned} \frac{1}{2^{2t}}\mathbb{E}(1 - \frac{\gamma}{2})^{2t} &= \frac{1}{2\pi 2^{2t}} \int_{-2}^2 (1 - \frac{\gamma}{2})^{2t} \sqrt{(4 - \gamma^2)} d\gamma \\ &= \frac{1}{2\pi 4^{2t}} \int_{-2}^2 (2 - \gamma)^{2t+1/2} \sqrt{2 + \gamma} d\gamma. \end{aligned}$$

By changing the integration variable to $\omega = \sqrt{2 + \gamma}$ we have

$$\begin{aligned}
\frac{1}{2^{2t}} \mathbb{E} \left(1 - \frac{\gamma}{2} \right)^{2t} &= \frac{1}{2\pi 4^{2t}} \int_{-2}^2 (2 - \gamma)^{2t+1/2} \sqrt{2 + \gamma} \\
&= \frac{1}{2\pi 4^{2t}} \int_0^2 \omega^2 (4 - \omega^2)^{2t+1/2} d\omega \\
&\stackrel{1}{=} \frac{2}{2\pi 4^{2t} (2t + 3/2)} \int_0^2 (4 - \omega^2)^{(2t+3/2)} d\omega \\
&\stackrel{2}{=} \frac{4^{2t+3/2}}{\pi 4^{2t} (2t + 3/2)} \int_0^{\pi/2} \cos^{4t+4} \theta d\theta \\
&= \frac{4}{\pi(t + 3/4)} \int_0^{\pi/2} \cos^{4t+4} \theta d\theta.
\end{aligned}$$

(1) is the result of integration by parts for $u = \omega$ and $dv = \omega(4 - \omega^2)^{2t+1/2} d\omega$. (2) is the result of another change of variable $\omega = 2 \cos(\theta)$. For the last integral we can use integration by parts to derive $\int_0^{\pi/2} \cos^{4t+4}(\theta) d\theta = \frac{4t+3}{4t+4} \int_0^{\pi/2} \cos^{4t+2}(\theta) d\theta$ and therefore, $\int_0^{\pi/2} \cos^{4t+4}(\theta) d\theta = \frac{(4t+3)(4t+1)\dots 1}{(4t+4)(4t+2)\dots 2} \frac{\pi}{2}$. Figure 7.1 compares the theoretical value predicted by the above formula with the result of Monte Carlo simulations on a medium size problem.

There are a couple of points about the above example that are worth mentioning:

Relatively fast convergence: It is clear that as t grows, the algorithm convergence is $O(1/t)$. This is faster than the rate predicted by the deterministic framework.¹

Coefficient universality: One of the remarkable properties of the above algorithm is that the convergence does not depend on the distribution of the coefficients and we only need $\mathbb{E}_F(X) = 0, E_F(X^2) = 1$. Therefore on a wide range of distributions the performance of the algorithm is the same. We call this phenomenon ‘coefficient universality’ of the above algorithm.

¹Since the maximum eigenvalue is converging to one, the problem is not strongly convex and therefore, the minimax mean square error at iteration $t < n/2$ is lower bounded by $\|x^0 - x^*\|_2/5$. For more information, see [83]

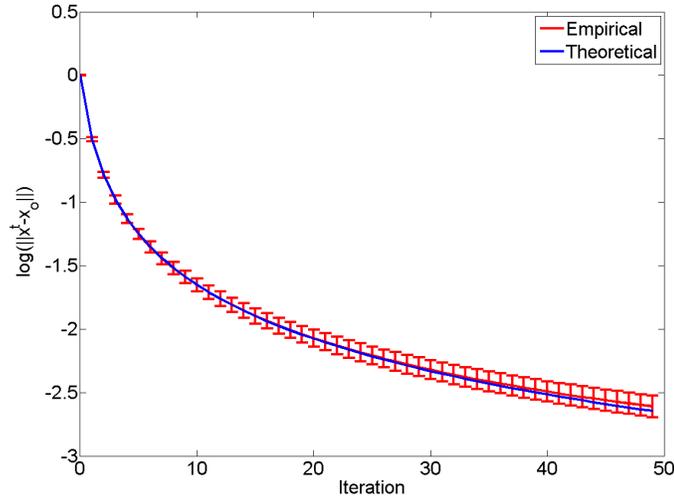


Figure 7.1: Comparison of the theoretical asymptotic prediction provided in Example 7.1 for a linear system and the Monte Carlo simulation result on a medium size problem, $n = 1000$. Elements of x_o are iid $N(0, 1)$, 1000 Monte Carlo simulations are considered. The bars show 95 percent confidence intervals.

Matrix universality: Although the matrix is drawn at random from $N(0, 1/n)$, all the arguments can be extended to more general distributions with no effort. For example, the semicircle law holds for all the distributions with $E_F(X^2) = 1/n$ and $\mathbb{E}_F(X) = 0$ and the sub-gaussianity argument is also true for any sub-gaussian matrix [108]. In other words, for a wide range of matrix ensembles the performance of the algorithm is the same. This phenomenon is called ‘matrix universality’.

As summarized in the above example the statistical approach for analyzing sparse recovery algorithms is more suitable for compressed sensing problems. However, unfortunately except for the AMP algorithm, the main focus has been on the deterministic convergence rates of different algorithms. In the next section, we discuss the statistical convergence rates for the ℓ_1 -minimization problem.

7.2.3 Average case convergence for CS problems

In the average case performance analysis, we exploit the randomness of A and instead of calculating $\|x^t - x_o\|_2^2$ on the worst possible case we consider $\mathbb{E}(\|x^t - x_o\|_2^2)$. Since we are interested in very high dimensional problem sizes, we consider the convergence of $\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}(\|x^t - x_o\|_2^2)$ while $\delta = n/N$ and $\rho = k/n$ are fixed. In order to show the strength of this type of analysis we mention the following theorem for the AMP algorithm from [38],[4].

Theorem 7.2.4. [38],[4] *Suppose that the elements of the measurement matrix are iid $N(0, 1/n)$. For $\delta \in [0, 1]$, $\rho < \rho_{SE}(\delta)$, the formal MSE of optimally-tuned AMP evolves to zero under SE. Furthermore, there exists $b > 0$ with the following property.*

$$\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E} \|x^t - x_o\|_2^2 \leq e^{-bt} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E} \|x^0 - x_o\|_2^2.$$

In other words, although AMP is a first-order method it converges exponentially fast in the MSE sense. As mentioned before the AMP algorithm matches the statistical framework we proposed in the last section and therefore we could analyze it theoretically. However there is not much information about the statistical convergence of FISTA, IST and other first-order methods. In this thesis, we use the tools we developed in this thesis in addition to an empirical approach to analyze these algorithms. We will first analyze the effects of coefficient ensemble and matrix ensemble on the performance of different algorithms. The last section will be devoted to the comparison of the average performance of different first-order methods.

7.3 Difficulty of compressed sensing problems

We observed that in the specific problem mentioned in Example 7.1, the mean square error at iteration t did not depend on the matrix ensemble or coefficient ensemble. The goal of this section is to discuss the effects of matrix distribution and coefficient distribution on the performance of several popular first-order methods for compressed sensing.

7.3.1 Notations

We consider the sparse recovery problem in the presence of noise. Let $y = Ax_o + w$, where x_o is the signal to be recovered, and w is iid $N(0, \omega)$. We denote a problem instance by $\Theta = (D_A, G, \epsilon; \delta, \nu)$. D_A represents the distribution from which the matrix is drawn, G is the probability density function of non-zero elements of x_o^2 , and ϵ is the probability that an element of x_o is non-zero. In other words, $x_{o,i} \sim (1 - \epsilon)\delta_0(x_{o,i}) + \epsilon G(x_{o,i})$, where $x_{o,i}$ is the i^{th} element of the vector x_o . Finally $\delta = n/N$ and $\nu = w/\mathbb{E}_G(X^2)$. Suppose that x^t is a sequence resulting from one of the algorithms. We define the mean-time-to-converge as $t(\alpha) \triangleq \inf\{t_0 : \lim_{N \rightarrow \infty} \mathbb{E}\|x^t - x_o\|_2^2 / \mathbb{E}\|x_o\|_2^2 \leq \alpha \quad \forall t > t_0\}$. $t(\alpha)$ depends on both the problem instance and the algorithm. It is also worth mentioning that if x^t does not converge to x_o in the mean square sense, then $t(\alpha)$ will be infinite for $\alpha < \lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \mathbb{E}\|x^t - x_o\|_2^2 / \mathbb{E}\|x_o\|_2^2$. Also, according to [38], $\lim_{N \rightarrow \infty} \mathbb{E}\|x^t - x_o\|_2^2$ for these problem instances may converge to their final value exponentially fast (linear convergence).

7.3.2 Matrix universality

In Example 7.1 we exhibited an algorithm whose performance was the same for a large class of random matrices. In this section our goal is to check the same phenomena for FISTA, IST, and AMP. Our approach in this section is empirical, i.e. we run Monte Carlo simulations to check the matrix universality hypothesis. Our experiments confirm the matrix universality hypothesis as is summarized in this finding.

Empirical Finding 1: Suppose that the elements of $n \times N$ measurement matrix are chosen iid at random from a ‘well-behaved’ probability distribution.³ Furthermore, the non-zero elements of the vector x_o are sampled randomly from a given distribution G . The observed behavior of $\mathbb{E}\|x^t - x_o\|_2^2 / N$ for the FISTA algorithm (or IST or AMP) will exhibit the same behavior as the Gaussian ensemble with large N .

²It may be replaced with the distribution function as well. However, for the simplicity of notation we assume that the probability density function exists.

³We assume that $\mathbb{E}(A_{ij}) = 0$ and $\mathbb{E}(A_{ij}^2) = \frac{1}{n}$.

We used a vague statement of ‘well-behaved’ since the exact specifications of the universality class are not known yet. The class of ensembles on which we have tested the above hypothesis is summarized in Table 7.1. We performed extensive Monte Carlo simulations to check the matrix universality hypothesis. Figure 7.2 shows the result of Monte Carlo simulations on FISTA and IST. To see the statistical analysis, range of problem instances we checked, our methodology, in addition to the results of our simulations for other algorithms refer to Section 7.6.2.

Table 7.1: Matrix ensembles considered in the matrix universality hypothesis tests.

Name	Specification
RSE	iid elements equally likely to be $\frac{\pm 1}{\sqrt{n}}$
USE	iid elements $N(0, 1/n)$
TERN	iid elements equally likely to be $0, \sqrt{3/2n}, \sqrt{-3/2n}$
TERN0P6	iid elements taken values $0, \sqrt{5/2n}, -\sqrt{5/2n}$ with $P(0) = .6$

7.3.3 Coefficient ensemble

In Example 7.1 we also observed the ‘coefficient universality’ phenomena. Our goal here is to check the coefficient universality for FISTA, IST, and AMP algorithms. As before suppose that the elements of x_o are drawn iid from the distribution $(1 - \epsilon)\delta_o(x_i) + \epsilon G(x_i)$ and that $\mathbb{E}_G(X^2)$ is bounded. Since we have observed universality phenomena for matrix ensembles we just consider Gaussian measurement matrices where the elements are iid $N(0, \frac{1}{n})$. Finally we use the notation $F_{\epsilon, \gamma}(\mu) = (1 - \epsilon)\delta_0(\mu) + \epsilon\delta_\gamma(\mu)$ and $G_\epsilon(x_i) = (1 - \epsilon)\delta_o(x_i) + \epsilon G(x_i)$. There are two factors that change the distribution here, ϵ , the fraction of non-zeros and G , the distribution on the non-zero elements. In this section we assume that ϵ is fixed and we just consider different G . Next section will be devoted to the analysis of ϵ . Before we see if ‘coefficient universality’ holds or not, let us explain formally what we mean by equivalence here.

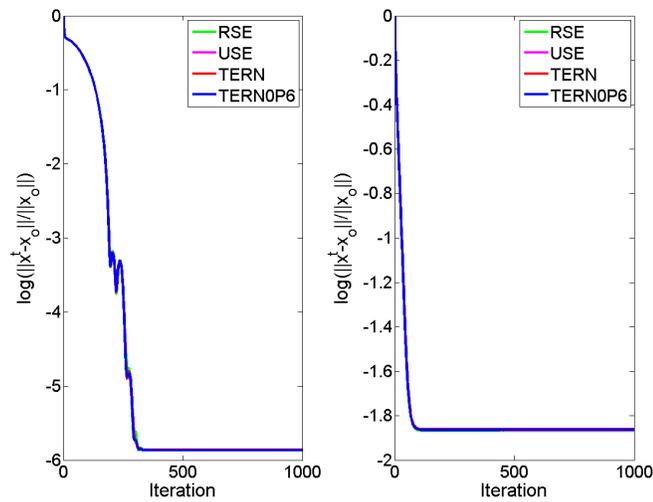


Figure 7.2: Checking the matrix universality hypothesis. Left: logarithm of the mean square error of FISTA for four different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .001$. Right: logarithm of the mean square error of IST for four different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .1$. The four curves shown in the above figures are on top of each other and this confirms the matrix universality hypothesis. For the statistical analysis of the discrepancies refer to Section 7.6.2.

Definition 7.3.1. Coefficient distributions Λ_ϵ and $G_{\epsilon'}$ are called equivalent for an iterative LASSO solver \mathcal{A} if and only if, for every $\lambda_{\Lambda,\epsilon}$ there exists $\lambda_{G,\epsilon'}$ such that for every $\alpha > 0$, $t_{\lambda_{\Lambda,\epsilon}}(\alpha) = t_{\lambda_{G,\epsilon'}}(\alpha)$ for the estimates of \mathcal{A} on the problem instances $(N(0, 1/n), \Lambda, \epsilon, \lambda_{\Lambda,\epsilon}; \delta, \nu)$ and $(N(0, 1/n), G, \epsilon', \lambda_{G,\epsilon'}; \delta, \nu)$ respectively and vice versa, i.e., for any $\lambda_{G,\epsilon'}$ there exists $\lambda_{\Lambda,\epsilon}$ with the same property.

Definition 7.3.2. Coefficient distributions Λ_ϵ and $G_{\epsilon'}$ are called equivalent for an AMPT if and only if, for every $\tau_{\Lambda,\epsilon}$ there exists $\tau_{G,\epsilon'}$ such that for every $\alpha > 0$, $t_{\tau_{\Lambda,\epsilon}}(\alpha) = t_{\tau_{G,\epsilon'}}(\alpha)$ for the estimates of AMPT on the problem instances $(N(0, 1/n), \Lambda, \epsilon, \tau_{\Lambda,\epsilon}; \delta, \nu)$ and $(N(0, 1/n), G, \epsilon', \tau_{G,\epsilon'}; \delta, \nu)$ respectively and vice versa, i.e., for any $\tau_{G,\epsilon'}$ there exists $\tau_{\Lambda,\epsilon}$ with the same property.

Definition 7.3.3. Coefficient Distribution Universality Hypothesis- Algorithm A satisfies the coefficient distribution universality hypothesis on a class of distribution \mathcal{C} , if and only if all any two distributions in \mathcal{C} are equivalent for A .

Lemma 7.3.1. For a fixed G with $E_G(X^2)$ bounded and for any α , all the coefficient ensembles of the form $(1 - \epsilon)\delta_o(\mu) + \epsilon|\alpha|G(\alpha\mu)$ are equivalent for FISTA, IST, and AMPT.

The proof is very simple and skipped.

Lemma 7.3.2. The coefficient universality hypothesis does not hold for FISTA, IST, or AMPT algorithms on $\mathcal{C} = \{U1, 3P, G1\}$ defined in table 7.2.

The proof may be found in Section 7.6.3.

Since the coefficient universality hypothesis does not hold for any of the sparse recovery algorithms, the next question is which distributions are less favorable for these algorithms.

Definition 7.3.4. Coefficient distribution Λ_ϵ is called less favorable than distribution $G_{\epsilon'}$ for LASSO solver \mathcal{A} if and only if, for every $\lambda_{F,\epsilon}$ there exists $\lambda_{G,\epsilon'}$ such that for every $\alpha > 0$, $t_{\lambda_{\Lambda,\epsilon}}(\alpha) \geq t_{\lambda_{G,\epsilon'}}(\alpha)$ on the problem instances $(N(0, 1/n), \Lambda, \epsilon, \lambda_{\Lambda,\epsilon}; \delta, \nu)$ and $(N(0, 1/n), G, \epsilon', \lambda_{G,\epsilon'}; \delta, \nu)$ respectively.

Table 7.2: Coefficient ensembles considered in coefficient ensemble analysis experiments.

Name	Specification
3P	iid elements taking value 0, 1, -1 with $P(0) = 1 - \epsilon$
5P1	iid elements taking values 0, ± 1 , ± 5 with $P(1) = P(-1) = .3\epsilon$ and $P(5) = P(-5) = .2\epsilon$
5P2	iid elements taking values 0, ± 1 , ± 5 with $P(1) = P(-1) = .3\epsilon$ and $P(20) = P(-20) = .2\epsilon$
U1	iid $U[0, 4]$
U2	iid $U[0, 20]$
G1	iid $N(0, 4)$

Similarly for comparing distributions for AMPT algorithm, we should just substitute τ wherever there is λ in the above definition as we did for the equivalence definition.

As mentioned in the beginning of this section our goal here is to characterize the distribution G and we consider fixed ϵ . Therefore we drop ϵ in our notations. In order to make any comparison the first step is to find the corresponding λ_G . The following lemma helps us in this regard.

Lemma 7.3.3. *Consider the distribution G with $E_G(X^2) = 1$. For every $\lambda \geq 0$ and for any $\nu > 0$ if*

$$\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_{x_o} \|\hat{x}_\lambda - x_o\|_2^2 \leq \epsilon,$$

then there exists a corresponding λ_G such that,

$$\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_G \mathbb{E}_{x_o} \|\hat{x}_{\lambda_G} - x_o\|_2^2 = \lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_{x_o} \|\hat{x}_\lambda - x_o\|_2^2.$$

It is worth mentioning that the condition $\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_{x_o} \|\hat{x}_\lambda - x_o\|_2^2 \leq \epsilon$ means the final error of the algorithm is less than $\frac{1}{N} E \|x_o\|_2^2$ which is the error of the algorithm if we set the estimate to 0.

Empirical Finding 2: *Under the above assumptions the 3P is less favorable for FISTA, IST than the other ensembles defined in table 7.2.*

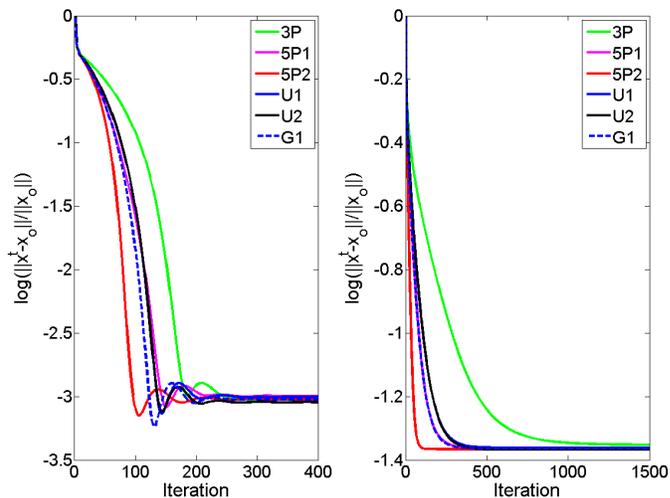


Figure 7.3: Logarithm of the relative mean square error for several distributions. 3P stands for three point prior or constant amplitude prior. The other abbreviations are explained in table 7.2. Left: $\delta = .5$, $\epsilon = .15$, $\lambda = .01$, $\sqrt{\nu} = .001$ and the algorithm is FISTA. Right: $\delta = .3$, $\epsilon = .03$, $\lambda = 0.01$, $\sqrt{\nu} = 0.01$ and the algorithm is IST. These simulations confirm the fact that 3P is less favorable than the other distributions.

Theorem 7.3.4. *3P is the least favorable distribution for AMP algorithm in the class $\mathcal{C} = \{F \mid \mathbb{E}_F(X^2) \leq C\}$.*

The proof of this theorem is sketched in Section 7.6.3.

Figure 7.3 summarizes the result of one of our experiments that led to the above finding. It should be mentioned that in these figures the final mean square error is not exactly the same for different distributions and therefore the definition may have been violated at the final iterations. Since the tuning has is done theoretically for the asymptotic setting, small discrepancies are expected here. However as the dimension of the problem increases, we expect the discrepancies to disappear. Refer to Chapter 6 for more information on this issue. The other problem instances we considered and the tuning step are explained in detail in Section 7.6.3.

7.3.4 Sparsity level

In the previous section we assumed that the sparsity level is fixed and discussed the distribution of the nonzero coefficients. In this section our goal is to see what happens if the distribution of non-zero coefficients is fixed and the sparsity level changes. The following lemma shows that again the ‘coefficient universality’ does not hold in this case.

Lemma 7.3.5. *$F_{\epsilon,1}$ and $F_{\epsilon',1}$ are not equivalent for any of the FISTA, IST and AMPT algorithms unless $\epsilon = \epsilon'$.*

The proof of the above lemma is summarized in Section 7.6.3. Although the above lemma is just considering three point priors, the extension to more general priors is very simple. For simplicity of notation we have mentioned the 3-point prior here. All the theorems mentioned in this section for $F_{\epsilon,1}$, can be easily extended to more general distributions G of non-zero coefficients. For more information on this you may refer to Section 7.6.4. Here since we fix the distribution of the non-zero coefficients we drop the dependence of λ on the distribution G and write shortly λ_ϵ .

Lemma 7.3.6. *Suppose that $\epsilon' \leq \epsilon$. For every $\lambda_\epsilon > 0$ and for any $\nu > 0$ if*

$$\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2 \leq 1.$$

there exists a corresponding $\lambda_{\epsilon'}$ such that,

$$\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2 = \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon',1/\sqrt{\epsilon'}}} \|\hat{x}_{\lambda_{\epsilon'}} - x_o\|_2^2$$

The proof of this lemma is sketched in Appendix 7.6.4. Note that in the above lemma the inputs are scaled such that the $\mathbb{E}\|x_o\|_2^2$ is fixed. This is important since we are dealing with the relative mean square error and that is what makes the above theorem non-trivial. Also the condition $\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2 \leq 1$ means that the algorithm is improving the estimate since 1 is the mean square error of the zero estimation. Figure 7.4 represents the values of λ that lead to the same relative mean square error on the phase plane.

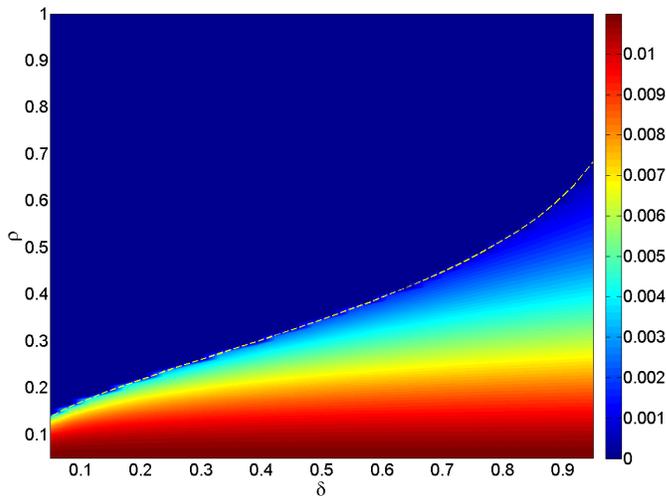


Figure 7.4: The values of λ that lead to the same relative mean square error. The elements of the measurement matrix are iid $N(0, 1/n)$ and the coefficients are drawn from 3-point prior $(1 - \epsilon)\delta_0 + \epsilon/2\delta_1 + \epsilon/2\delta_{-1}$. The standard deviation of the measurement noise is $\sigma = .001$. The dashed curve represents $.95\rho_{SE}$. The results are derived from the theoretical calculations.

Theorem 7.3.7. *If $\epsilon < \epsilon'$, $F_{\epsilon',1}$ is less favorable than $F_{\epsilon,1}$ for AMPT algorithm.*

The proof of this theorem is given in Section 7.6.4. Empirical observations confirm similar results for the FISTA and IST algorithms. These observations are summarized in the following.

Empirical Finding 3. For a fixed value of δ if $\epsilon < \epsilon'$, $F_{\epsilon,1}$ is more favorable for FISTA and IST than $F_{\epsilon',1}$.

One of the simulation results that confirms the above empirical finding is depicted in Figure 7.5. The values of λ are set according to Lemma 7.3.6. As before because of the finite sample size effect the hypothesis may seem to be violated at the final iterations. However, we expect the discrepancies to vanish as the dimension increases. For more information on the problem instances we have tested and on our methodology please refer to Section 7.6.4.

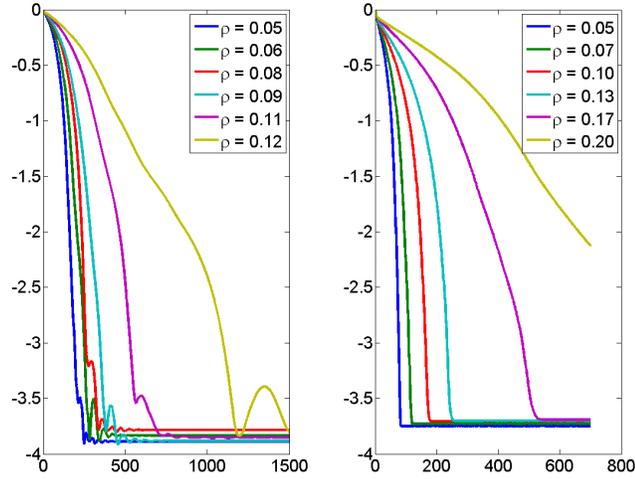


Figure 7.5: Comparison of convergence of FISTA and IST algorithms for different values of ρ . Left: $\delta = .05$, $N = 1500$, $\sigma = .001$ and the algorithm is FISTA. Right: $\delta = 0.32$, $N = 1500$, $\sigma = .001$ and the algorithm is IST. These simulation confirm the hypothesis that larger values of ϵ are less favorable.

7.4 Other first-order methods

So far we have mainly discussed three algorithms FISTA, IST, and AMP. However there are several other first-order algorithms proposed for solving the ℓ_1 minimization problem. Here we discuss some of these algorithms.

7.4.1 Regularization parameter and continuation strategies

So far we have considered three different factors, the distribution of the matrix, distribution of coefficients and the sparsity level. In this section, we analyze the last parameter, i.e. the regularization parameter. Our goal is to see how the convergence rate and $\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} E \|x^t - x_o\|_2^2$ changes with respect to the regularization parameter of the LASSO problem.

Lemma 7.4.1. *Suppose that the sparsity level $\rho < \rho_{SE}$ is given and $y = Ax_o + \sigma z$ where $z \sim N(0, 1)$. Let λ^* denote $\arg \min_{\lambda} \lim_{N \rightarrow \infty} \frac{1}{N} E \|x_{\lambda} - x_o\|_2^2$. For $\lambda > \lambda^*$ the error*

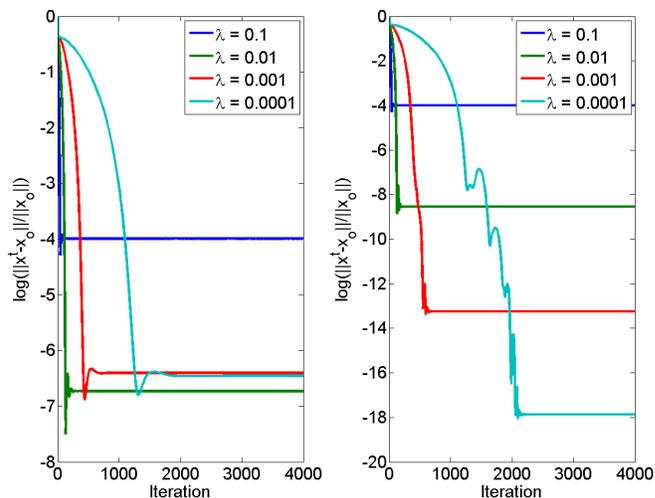


Figure 7.6: Performance of FISTA in terms of λ . Left: $\delta = .3$, $\sqrt{\nu} = .01$, $N = 2000$, $\rho = .5\rho_{Se}$; Right: the same problem with $\nu = 0$

$\lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E} \|\hat{x}_\lambda - x_o\|_2^2$ is monotonically increasing and below λ^* it is monotonically decreasing. As $\sigma \rightarrow 0$ the optimal λ also goes to zero.

The above phenomenon is shown in Figure 7.6. The simulation results also confirm the following observation that smaller values of λ result in more difficult problems. This observation has led the researchers to the idea of continuation strategies, i.e. starting from larger values of λ and solving the problem for those and then decrease the value of λ . One of the first algorithms that used the idea of continuation for the IST is [64] and is called FPC for fixed point continuation. This algorithm starts with a large value of λ_1 and uses IST to find \hat{x}_{λ_1} . After the convergence it switches to smaller value $\lambda_2 = \kappa \lambda_1$ for $\kappa < 1$ and initialize IST with \hat{x}_{λ_1} and again repeats the iterations. This process continues until it reaches the desired value of λ . As we will see in the simulation section this approach is extremely useful specially when λ is small compared to $\|A^*y\|_\infty$. For the application of continuation strategies in other algorithms you may refer to [49], [64], [96], [57], [12], [33], and [86]. Clearly, since at the heart of this algorithm lies iterative soft thresholding algorithm the problems that are more difficult for IST will be also difficult for the FPC algorithm.

7.4.2 Other Heuristics

So far we have explained four different first-order algorithms. AMP, IST, FISTA, and FPC. Several other heuristics have been also proposed for speeding up the IST algorithm for CS problems. We briefly review some of these algorithms. There is no proof either from the deterministic point of view nor from the statistical point of view to show that the following algorithms outperform IST, however in practice they usually work better.

7.4.3 TwIST

Inspired by two step methods for solving linear systems [1], Bioucas-Dias and Figueredo proposed TwIST or two stage iterative thresholding algorithm [9].

$$x^{t+1} = (1 - \alpha)x^{t-1} + (\alpha - \beta)x^t + \beta\eta(x^t + A^*(y - Ax^t); \lambda). \quad (7.15)$$

Similar to FISTA this algorithm uses longer history of the algorithm.

7.4.4 GPSR-BB

It is not difficult to see that by writing $x = u - v$ where $u \succeq 0$ and $v \succeq 0$ and by defining z as a concatenation of u and v , \mathcal{P}_λ can be cast as,

$$\begin{aligned} \min c^* z + \frac{1}{2} z^* B z. \\ z \succeq 0. \end{aligned} \quad (7.16)$$

Defining $b = A^*y$, B, c are,

$$B = \begin{pmatrix} A^*A & -A^*A \\ -A^*A & A^*A \end{pmatrix}, \quad c = \lambda \mathbf{1} + \begin{pmatrix} -b \\ b \end{pmatrix}.$$

Therefore the following first-order algorithm can be used for solving the above

problem,

$$\begin{aligned} w^t &= (z^t - \alpha^t(Bz^t - c))_+, \\ z^{t+1} &= z^t + \lambda^t(w^t - z^t). \end{aligned} \tag{7.17}$$

This algorithm was first proposed in [57] and was called Gradient Projection for Sparse Reconstruction or GPSR. GPSR-BB is the same algorithm with a subtle difference. It uses the Barzilai-Borwein idea [3] for setting the step size.

7.4.5 FPC-AS

This algorithm has been motivated by both the FPC algorithm and some other greedy algorithms such as StOMP or CoSaMP. As in FPC the algorithm starts at a large value of λ_1 and applies iterative soft thresholding algorithm to achieve \hat{x}_{λ_1} . The only difference from FPC is that the step size is chosen dynamically according to Barzilai-Borwein. After the convergence it then fixes the active set and the signs of the elements and replaces $\|\cdot\|_1$ -norm with s^*x where s is the sign of \hat{x}_{λ_1} . Then the ℓ_1 minimization problem is cast as a smooth optimization problem that can be solved by other iterative methods such as conjugate gradient or quasi Newton methods on the active set. After convergence $\lambda_2 < \lambda_1$ is considered in the optimization and \hat{x}_{λ_1} is used as the initialization in the IST algorithm. The algorithm iterates until it reaches the actual value of λ .

7.4.6 Properties of statistical convergence rates

Matrix universality

We did similar tests on FPC-AS, TWIST and GPSR and observed that the matrix universality hypothesis holds for these algorithms as well. For more information on the experiments we conducted for these algorithms you may refer to the Section 7.6.2.

Coefficient ensemble

We conducted similar experiments to analyze the effect of coefficient ensemble on the performance of FPC-AS, TWIST and GPSR as well. The conclusion here was the same as the our conclusion about FIST, IST and FPC. For more information about the tests refer to Section 7.6.3.

Sparsity level

Our experiments on TwIST, FPC-AS and GPSR confirmed that finding 3 holds for these algorithms as well. For more information you may refer to Section 7.6.4.

7.5 Comparison results

So far we have explained the statistical properties of different algorithms. The goal of this section is to compare the convergence of the algorithms explained above. The main point is that since the computational complexity of the iterations of each algorithm is different, counting the number of iterations does not provide a fair comparison. Therefore we calculate the CPU time that each algorithm takes to achieve a certain accuracy.

7.5.1 Exact recovery problem

In this section we consider the exact recovery problem. We therefore assume that the signal is exactly sparse and that $\rho \leq \rho_{SE}$. As mentioned before if we run our experiments on Gaussian matrix ensemble, according to the universality hypothesis the results for a wide class of distributions will be the same. Therefore we do the simulations on two different types of matrices. Gaussian and partial DCT. The partial DCT ensemble is constructed by selecting n rows of the DCT matrix at random and normalizing them such that each column has ℓ_2 norm equal to one. These measurement matrices are more useful in practice for two different reasons. First, applying

A and A^* to a vector needs $O(N \log N)$ operations rather than $O(nN)$ and therefore they are suitable for first-order methods. Second, we do not need to store the whole matrix in the memory.

Two different coefficient ensembles are considered in our simulations $3P$ and Cauchy. According to Section 7.3.3 we know that when we consider the relative mean square error $3P$ is less favorable than the other ensembles and that is why this ensemble is considered. The second ensemble is meant to test the performance of the algorithms on very high dynamic range signals. It seems that if we are interested in correct support recovery then these distributions may be less favorable than $3P$. For all these problem suites the performance of AMPM is compared with the other algorithms. AMPM is a version of the AMPT algorithm in which the parameter τ is set by the maximin tuning. For more information on this framework you may refer to Chapters 3 and 4. However, since the other algorithms are solving \mathcal{P}_λ and we know that the solution of \mathcal{P}_λ converges to the solution of \mathcal{Q}_0 as $\lambda \rightarrow 0$, we use the other algorithms to solve \mathcal{P}_λ for small values of λ . We consider three different values of λ : $.01\|A^*y\|_\infty$, $.001\|A^*y\|_\infty$ and $.0001\|A^*y\|_\infty$. In each case we report the final errors of each algorithm as well. The error that we are considering in these simulations is $\|x^t - x_o\|_2 / \|x_o\|_2$. In the tables we will use the following notations:

- $\text{fpe} = \log\left(\frac{\|x^\infty - x^*\|_2}{\|x^*\|_2}\right)$.
- $\iota_0 = \max\{t : \text{MDR}^t = 0 \text{ and } \text{DR}^t < \delta\}$.
- $\iota_1 = \inf\{t : \frac{\|x^{t'} - x^*\|_2}{\|x^*\|_2} \leq 0.01 \quad \forall t' > t\}$.
- $\iota_2 = \inf\{t : \frac{\|x^{t'} - x^*\|_2}{\|x^*\|_2} \leq 0.001 \quad \forall t' > t\}$.
- $\iota_3 = \inf\{t : \frac{\|x^{t'} - x^*\|_2}{\|x^*\|_2} \leq 0.0001 \quad \forall t' > t\}$.

In the above equations MDR stands for the missed detection rate and DR stands for the actual detection rate. We have introduced ℓ_0 since some of the algorithms use the de-biasing technique after the convergence of the algorithm. ι_0 is the first iteration at which an oracle can stop the algorithms and use de-biasing to get to the correct answer. However in the table we have used $\ell_0, \ell_1, \ell_2, \ell_3$ instead of $\iota_0, \iota_1, \iota_2, \iota_3$.

ℓ_i represents the average time spent by the algorithm until it reaches iteration ℓ_i . All the simulations are done on Pentium 4, 2.44 GHz Dual core, 4GByte RAM machine. For the calculation of fpe we ran FISTA for 5000 iterations and replace x^∞ with x^{5000} of FISTA in the calculation of fpe. The problem instances are chosen such that FISTA converges to its final solution in 5000 iterations. Finally each data point in the table is the average of 100 realizations of the same problem suite.

7.5.2 Measurement noise

In the last section we considered the exact recovery problem and compared a few different algorithms. The goal of this section is to evaluate the performance of different algorithms in the presence of measurement noise. Again we just consider the Gaussian and partial DCT matrices. We fix $N = 2048$ and $\delta = .25$. Three different values of ρ are considered as in the previous section. The measurements are now corrupted with the noise $y = Ax_o + \sqrt{\nu}n$. We consider $\sqrt{\nu} \in \{0.001, .01, .1\}$. The goal is still to recover x_o . $\lambda = \sqrt{\nu}\mathbb{E}\|A^*n\|_\infty$ is chosen for the ℓ_1 solvers. This ensures that x_o is still in the feasible set. AMPM is still used in this case. The other parameters of all these algorithms are set to the default values as explained in the previous section.

7.6 Detailed experiments and proofs

7.6.1 AMPT-LASSO calibration

Here we briefly summarize some of the recent advances in predicting the asymptotic performance of the LASSO algorithm. In chapter 4 we introduced the AMPT algorithm. Starting at $x^0 = 0$, it proceeds according to the iteration:

$$\begin{aligned} z^t &= y - Ax^t + \frac{|I^t|}{n} z^{t-1}, \\ x^{t+1} &= \eta(A^* z^t; \tau \hat{\sigma}^t). \end{aligned}$$

As mentioned in that chapter, state evolution provides a framework for predicting the asymptotic performance of the AMPT algorithm. According to this framework, if m_t

is the normalized mean square error at iteration t the mean square error at iteration $t + 1$ is

$$m_{t+1} = \Psi(m_t),$$

where

$$\Psi(m_t) \triangleq \mathbb{E}(\eta(X + \sqrt{\frac{m_t}{\delta} + \nu}Z; \tau\sqrt{\frac{m_t}{\delta} + \nu}) - X)^2.$$

$X \sim (1 - \epsilon)\delta_0(\mu) + \epsilon G(\mu)$ and $Z \sim N(0, 1)$ are two independent random variables. Clearly, the final mean square error of the AMPT algorithm (as $N \rightarrow \infty$) corresponds to the stable fixed points of the Ψ function. It is proved that the Ψ function is concave and therefore it has just one stable fixed point. In addition to MSE, other observables of AMPT algorithm can be calculated through the state evolution framework. Some examples are the following:

- Equilibrium threshold

$$\theta_\infty = \tau\sqrt{\frac{m_\infty}{\delta} + \nu}.$$

- Equilibrium Detection Rate

$$\text{EqDR} = \mathbb{P}\{|\eta(Y_\infty; \theta_\infty)| \neq 0\},$$

$$\text{where } Y_\infty = X + \sqrt{\frac{m_\infty}{\delta} + \nu}Z.$$

m_∞ represents the fixed point of Ψ function. The final component that will be used in our arguments is the equivalence between LASSO and AMPT solutions. The following finding taken from [41], which has been recently proved in case of Gaussian measurement matrices in [4], explains the equivalence.

Theorem 7.6.1. [41], [4] *For each $\lambda \in [0, \infty)$ we find that $\text{AMPT}(\tau(\lambda))$ and $\text{LASSO}(\lambda)$ have statistically equivalent observables. In particular the MSE have the same value when τ and λ satisfy the following relation:*

$$\lambda = \theta_\infty(\tau)(1 - \text{EqDR}(\tau)/\delta).$$

Lemma 7.6.2. [41] Define τ^0 , so that $\text{EqDR}(\tau) \leq \delta$ when $\tau > \tau^0$. For each λ there is a unique value of $\tau(\lambda) \in [\tau_0, \infty)$ such that

$$\lambda = \theta_\infty(\tau)(1 - \text{EqDR}(\tau)/\delta).$$

The above discussion is mainly used in the calibration of λ for two different distributions.

7.6.2 Discussion of matrix universality

Data generation

To check the universality with respect to the matrix ensemble we considered four different distributions and for each of them we tried different combinations of the other parameters. The matrix ensembles we considered are $\mathcal{M} = \{\text{RSE}, \text{USE}, \text{TERN}, \text{TERN0P6}\}$. These ensembles are explained in table 7.1. For each matrix ensemble we considered 36 different problems that are formed as a combination of the following values for δ , ρ , λ , and ν .

- $\delta \in \{.2, .5, .7\}$
- $\rho \in \{.25\rho_{\text{SE}}, .5\rho_{\text{SE}}, .75\rho_{\text{SE}}\}$.
- $\lambda \in \{.01, .001\}$.
- $\sigma \in \{0, .01\}$.

In the above experiments ± 1 coefficient ensemble is considered. We also chose 5 random subsets of the above instances and tested them on Uniform $[0, 1]$ and $N(0, 1)$ coefficient ensembles. We ran these algorithms FISTA, IST, TwIST, FPC-AS, and GPSR on the problem instances for a number of iterations specified in table 7.3. We considered 80 Monte Carlo samples for each problem instance and each algorithm.

Table 7.3: Number of iterations of each algorithm in matrix universality experiments.

Algorithm	Number of Iterations
FISTA	2000
IST	6000
GPSR	2000
TWIST	2000
FPC-AS	1000

Testing universality hypothesis

Our hypothesis which was explained in Section 7.3.2 is as follows.

Empirical Finding 1: Suppose that the elements of the $n \times N$ measurement matrix are chosen iid at random from a “well-behaved” probability distribution⁴. Furthermore, the non-zero elements of the vector x_o are sampled randomly from a given distribution G . The observed behavior of $\mathbb{E}\|x^t - x_o\|_2^2/N$ for the FISTA algorithm (or IST,AMP, FPC, GPSR, TWIST, FPCAS) will exhibit the same behavior as the Gaussian ensemble with large N .

Our goal is to empirically test this hypothesis and see if the experiments confirm or reject the hypothesis

$$H_0 : MSE_1^t = MSE_2^t \quad \text{versus} \quad H_1 : MSE_1^t \neq MSE_2^t. \quad (7.18)$$

In the experimental setup we chose the sample size equal to 80. Since the sample size is large enough we use the two sample Z-test to test the universality hypothesis. If \widehat{MSE}_1^t is the mean square error for the Gaussian ensemble at iteration t and \widehat{MSE}_2^t the MSE of another ensemble at the same iteration the Z-score is defined as,

$$Z(\widehat{MSE}_1^t, \widehat{MSE}_2^t) = \frac{\widehat{MSE}_1^t - \widehat{MSE}_2^t}{SD(\widehat{MSE}_1^t - \widehat{MSE}_2^t)} \quad (7.19)$$

⁴We assume that $\mathbb{E}(A_{ij}) = 0$ and $\mathbb{E}(A_{ij}^2) = \frac{1}{n}$.

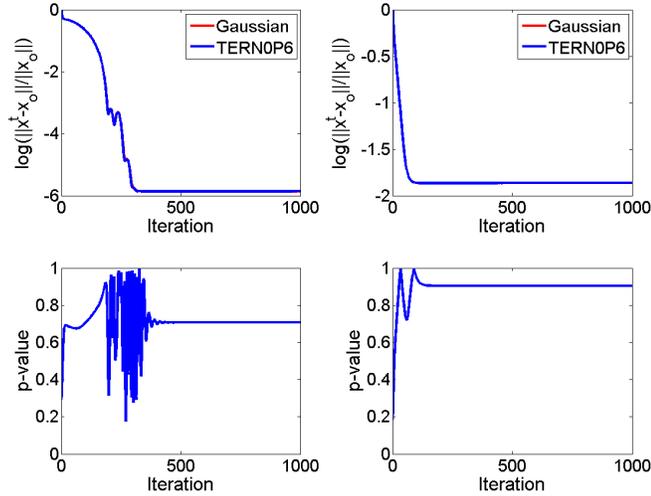


Figure 7.7: Checking matrix universality hypothesis. Top-Left: logarithm of the mean square error of FISTA for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .001$. Bottom-Left: The p-values of the test specified in (7.18). Top-Right: logarithm of the mean square error of IST for two different matrix ensembles defined in Table 7.1 at $N = 2000$, $\delta = .5$, $\rho = .25\rho_{SE}$, $\lambda = .1$. Bottom-right: The p-values of the test specified in (7.18). Clearly the p-values are large for both experiments and therefore we cannot reject the null hypothesis.

Under the universality hypothesis Z has approximately Gaussian distribution. We use this distribution to calculate the p-values. These p-values are displayed in Figures 7.7, 7.8. In case of FPCAS the match does not seem as good as other ensembles. However this is due to large variances of our estimations as explained by the p-values. We also considered the same problem instances for AMPT algorithm. The only difference is that for AMPT we considered $\tau \in \{1.5, 2\}$.

7.6.3 Discussion of coefficient distribution experiments

Tuning process

Let the elements of A be iid $N(0, 1/n)$ and the elements of the measurement vector are also drawn iid from $(1 - \epsilon)\delta_0(x_{o,i}) + \epsilon\delta_1(x_{o,i})$. For a fixed value of λ and ν we

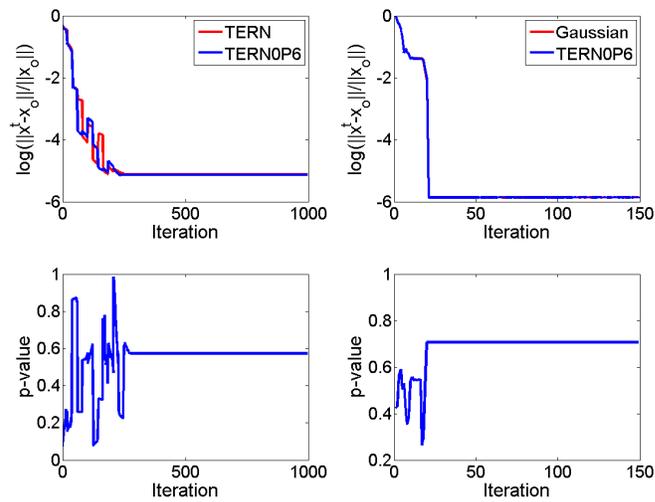


Figure 7.8: Checking matrix universality hypothesis of FPC-AS. Top-left: logarithm of the mean square error for two different matrix ensembles defined in Table 7.1 at $N = 2000, \delta = .5, \rho = .75\rho_{SE}, \lambda = .001, \nu = 0$. Bottom-left: The p-values of the test specified in (7.18). Top-right: logarithm of the mean square error for two different matrix ensembles defined in Table 7.1 at $N = 2000, \delta = .5, \rho = .25\rho_{SE}, \lambda = .001, \nu = 0$. Bottom-right: The p-values of the test specified in (7.18). Clearly the p-values are large for both experiments and therefore we cannot reject the null hypothesis.

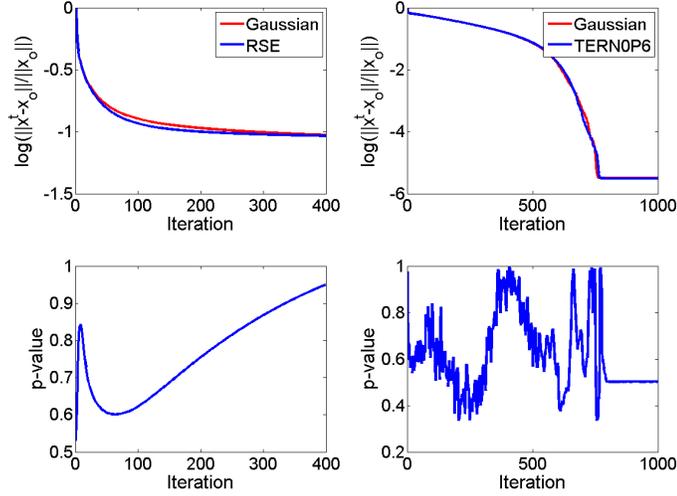


Figure 7.9: Matrix universality results for Left: AMPT with $\delta = .5$, $\rho = .5\rho_{SE}$, $\tau = 2$, $\nu = 0$. Right: TWIST with $\delta = .5$, $\rho = .5\rho_{SE}$, $\lambda = .001$, $\nu = 0$. In both simulations the number of Monte Carlo samples is 80.

calculate $\lim_{N \rightarrow \infty} \frac{1}{N} \|\hat{x}_\lambda - x_o\|_2^2$. According to Section 7.6.1, we can calculate this error theoretically. Now, suppose that we change the distribution of x_o to $(1 - \epsilon)\delta_0(x_{o,i}) + \epsilon G(x_i)$ where $E_G(X^2) = 1$. The goal is to find λ_G such that it may lead to the same value of asymptotic mean square error. In this section we answer two main questions. 1. Does such λ_G always exist? 2. If it exists, how can we calculate λ_G ? The key part in answering these two questions is the connection between LASSO and AMPT. Let us first prove the following lemma which was also stated in the text.

Lemma 7.6.3. *Consider the distribution G with $E_G(X^2) = 1$. For every $\lambda \geq 0$ and for any $\nu > 0$ if*

$$\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_{x_o} \|\hat{x}_\lambda - x_o\|_2^2 \leq \epsilon,$$

then there exists a corresponding λ_G such that,

$$\lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_G \mathbb{E}_{x_o} \|\hat{x}_{\lambda_G} - x_o\|_2^2 = \lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_{x_o} \|\hat{x}_\lambda - x_o\|_2^2,$$

Proof. According to Lemma 7.6.2, there exist a value of τ for which the asymptotic

mean square error of the state evolution frameworks is the same as the asymptotic mean square error of LASSO(λ). Here is our approach for solving this lemma. The first thing that we want to prove is that there exists a value of $\underline{\tau}$ for which the asymptotic mean square error of AMPT($\underline{\tau}$) on G is below the MSE of AMPT(τ) on $F_{\epsilon,1}$. Then we prove there exists a value of $\bar{\tau}$ for which the asymptotic mean square error of AMPT($\bar{\tau}$) is larger than the mean square error of AMPT(τ) on $F_{\epsilon,1}$. Finally we will use implicit function theorem to prove that there exists a value of τ_G for which the asymptotic mean square error of AMPT(τ_G) on G is exactly the same as the MSE of AMPT(τ) on $F_{\epsilon,1}$. Choice $\bar{\tau} = \infty$ is clear. Therefore we focus on constructing a choice for $\underline{\tau}$. The claim is that $\underline{\tau} = \tau$.

Claim 1: For every m and for any β we have, $\mathbb{E}_G \mathbb{E}_X (\eta(X + \sqrt{m/\delta}Z; \beta) - X)^2 \leq \mathbb{E}_{\delta_1} \mathbb{E}_X (\eta(X + \sqrt{m/\delta}Z; \beta) - X)^2$.

First we should emphasize that once we prove the above claim, since the Ψ function is equal to

$$\begin{aligned} \Psi_{G_\epsilon}(m) &= (1 - \epsilon) \mathbb{E}(\eta(\sqrt{\frac{m}{\delta}} + \nu; \tau \sqrt{m/\delta + \nu}))^2 \\ &\quad + \epsilon \mathbb{E}_G \mathbb{E}_X (\eta(X + \sqrt{m/\delta + \nu}Z; \tau \sqrt{m/\delta + \nu}) - X)^2, \end{aligned}$$

from the above claim we will have,

$$\Psi_{G_\epsilon}(m) \leq \Psi_{F_{\epsilon,1}}(m),$$

and therefore the fixed point for G will happen at a lower value of m . Therefore we just have to prove the above claim. By simple calculations we can show that,

$$\frac{d^2}{(dX^2)^2} (\eta(X + \sigma Z; \beta) - X)^2 = \frac{1}{-\sigma X} (\phi((\beta - X)/\sigma) + \phi((\beta + X)/\sigma)) < 0.$$

Using this concavity in addition to Jensen inequality it is very easy to prove our claim. So far, we have proved the existence of $\underline{\tau}$ and $\bar{\tau}$. The final step is to prove continuity

of the fixed point of Ψ in terms of τ . This is an easy implication of the implicit function theorem and therefore our claim is proved. Now that we could find τ_G we can again use Lemma 7.6.2 to get λ_G . \square

The interesting fact about the above proof is that it is also kind of constructive and therefore we use the same approach for calculating λ_G . In other words, we first find the value of τ that corresponds to λ and then we find the value of τ_G that gives the same MSE as τ on G and finally we calculate λ_G from τ_G .

Note: There might be more than one value of λ that generates the same mean square error on $F_{\epsilon,1}$ and there may be more than one value of λ_G with the same mean square error. In these cases we compare the fastest achievable rates for each case and as was shown in Section 7.4.1 this corresponds to comparison of largest λ and largest λ_G .

Theorem 7.3.4 *\mathcal{P} is the least favorable distribution for AMP algorithm in the class $\mathcal{C} = \{F \mid E_F(X^2) \leq C\}$.*

Proof. Suppose that an arbitrary distribution $G \in \mathcal{C}$ is given. According to Lemma 7.3.1 without loss of generality we assume $E_G(X^2) = 1$. Our goal is to prove that $F_{\epsilon,1}$ is less favorable than G_ϵ . We choose $\tau_G = \tau$. Let MSE_G^t the mean square error on G_ϵ problem instance at time t and MSE_F^t represent the same thing for $F_{\epsilon,1}$. Suppose that $MSE_G^t \leq MSE_F^t$ at time t , our goal is to first prove that $MSE_G^{t+1} \leq MSE_F^{t+1}$.

$$MSE_G^{t+1} = \mathbb{E}_{G_\epsilon} \mathbb{E}_X (\eta(X + \sqrt{\frac{MSE_G^t}{\delta} + \nu} Z; \tau \sqrt{\frac{MSE_G^t}{\delta} + \nu}) - X)^2;$$

Also let us define,

$$\tilde{MSE}_F^{t+1} = \mathbb{E}_{F_{\epsilon,1}} \mathbb{E}_X (\eta(X + \sqrt{\frac{MSE_G^t}{\delta} + \nu} Z; \tau \sqrt{\frac{MSE_G^t}{\delta} + \nu}) - X)^2;$$

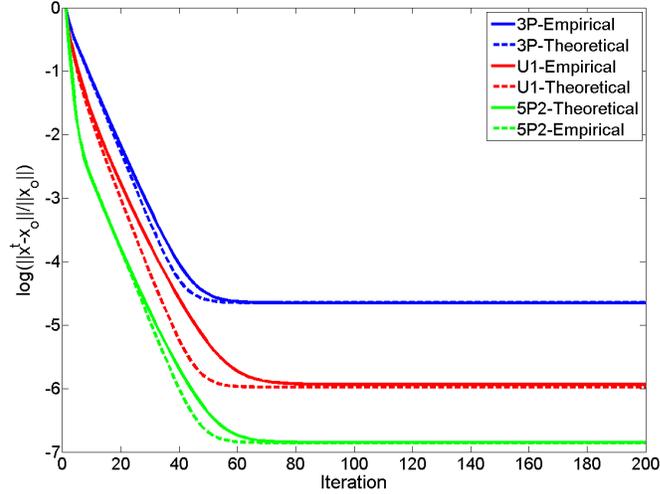


Figure 7.10: Performance of AMPT algorithm on a few different distributions. In this simulation $N = 4000$, $\delta = .5$, $\epsilon = .075$, $\tau = 2$, $\sqrt{\nu} = .001$ and the number of Monte Carlo samples 25. For more information on the decrease of the discrepancy between theory and Monte Carlo simulation refer to [41].

Our claim is that,

$$MSE_G^{t+1} \leq \tilde{MSE}_F^{t+1} \leq MSE_F^{t+1}.$$

The second inequality is a simple result of the fact that the mean square error is non-decreasing function of standard deviation of the noise Z . The first inequality is the result of Jensen inequality. In the last lemma we proved that $\mathbb{E}_X(\eta(X + \sqrt{\frac{MSE_G^t}{\delta} + \nu}Z; \tau\sqrt{\frac{MSE_G^t}{\delta} + \nu}) - X)$ is a concave function of X ; The proof of the main theorem is now a very simple induction. Since the mean square error for the two distributions are the same at iteration one. The proof is therefore complete. \square

Figure 7.10 shows the result of the above theorem in practice.

Lemma 7.3.2 *The coefficient universality hypothesis does not hold for FISTA, IST, or AMPT algorithms on $\mathcal{C} = \{U1, 3P, G1\}$.*

Proof. The proof is based on some of the facts that were proved in the above theorem. Suppose that two distributions F and G are equivalent for AMPT. Then we can conclude that for every τ_G there exists a τ_F such that AMPT may lead to the same relative mean square error on problem instances $(N(0, 1/n), F, \epsilon, \tau_F; \delta, \sigma)$, $(N(0, 1/n), G, \epsilon, \tau_G; \delta, \sigma)$. Now suppose that $G = N(0, 1)$ and $F = \frac{1}{2}\delta_1 + \frac{1}{2}\delta_{-1}$. For a given noise variance $\nu > 0$ set τ_G such that it minimizes the mean square error for $G = N(0, 1)$. We claim that no τ_F leads to the same mean square error for F . Suppose that such τ_F does exist; According to the previous theorem and strict concavity of $\mathbb{E}_X(\eta(X + \sigma Z; \tau\sigma) - X)^2$ with respect to X , the final mean square error of τ_F on G is less than the mean square error on $3P$ itself. Since we assumed that the final mean square on F is the same as the minimum mean square error on G this is a contradiction. Therefore there is no equivalent τ_F that may lead to the same mean square error.

For the Lasso solvers we use the connection between the fixed points of Lasso and AMPT and the above observation. Again we consider the value of λ_G that leads to the minimum mean square error and with a very similar argument claim that there is no λ_F with the same mean square error. \square

Experimental setup

Since we have demonstrated the universality hypothesis on the matrix ensemble, in these experiments we just consider the Gaussian matrix. In order to analyze the performance of different algorithm we considered the coefficient ensembles specified in table 7.2. For each algorithm and each coefficient distribution we considered the following problem instances,

- $\delta = .5, \epsilon = .15, \lambda = .01, \sigma = .001$
- $\delta = .5, \epsilon = .075, \lambda = .001, \sigma = .001$
- $\delta = .3, \epsilon = .07, \lambda = .01, \sigma = .001$
- $\delta = .3, \epsilon = .03, \lambda = .001, \sigma = .001$

- $\delta = .5$, $\epsilon = .03$, $\lambda = .001$, $\sigma = .01$

We considered 4000 iterations of the FISTA, 2000 iteration of FPC-AS, 8000 iterations of TWIST, IST, GPSR which are enough for convergence of all the above cases. Finally the results are the average of 80 Monte Carlo simulations. We also considered the following values of α in the calculation of $t_\lambda(\alpha)$, where $\alpha \in \{.2, .1, .05, .02, .01, .005, .002, .001, .0005\}$.

Designing a test

As mentioned before the goal is to compare $t(\alpha)$ for a given distribution versus $t(\alpha)$ at $F_{\epsilon,1}$. Since in our experiments $t(\alpha)$ is calculated from the average of Monte Carlo samples we assume that the sample size is large enough and we can assume they are samples from Gaussian normal distributions, i.e.,

$$\begin{aligned} t_\lambda(\alpha) &\sim N(\mu_F, \sigma_F^2), \\ t_{\lambda_G}(\alpha) &\sim N(\mu_G, \sigma_G^2). \end{aligned}$$

The goal is to check the null hypothesis

$$H_0 : \mu_G \leq \mu_F \text{ versus the alternative } H_1 : \mu_G > \mu_F. \quad (7.20)$$

Our claim is that the uniformly most powerful (UMP) test exists for this test and is given according to $t_\lambda(\alpha) - t_{\lambda_G}(\alpha) > \omega$ for some ω that satisfies the level constraint.

Lemma 7.6.4. *For the above test, $t_\lambda(\alpha) - t_{\lambda_G}(\alpha) > \omega$ is the UMP test with level constraint γ , if and only if ω is chosen to satisfy the level constraint.*

Proof. The proof is standard in the statistical testing literature. We first consider a specific choice of an alternative, let's say (μ'_G, μ'_F) , and test it versus the whole null. To do this test we consider the least favorable distribution on the null space. It is easy to confirm that the least favorable distribution is a point mass on $(\frac{\mu'_F \sigma_G^2 + \mu'_G \sigma_F^2}{\sigma_F^2 + \sigma_G^2}, \frac{\mu'_F \sigma_G^2 + \mu'_G \sigma_F^2}{\sigma_F^2 + \sigma_G^2})$. Now that the least favorable distribution has such a simple form one can easily use Neyman-Pearson lemma to derive the UMP test. Since this test does not depend on the choice of (μ'_G, μ'_F) , it is the most powerful test for the original problem as well. \square

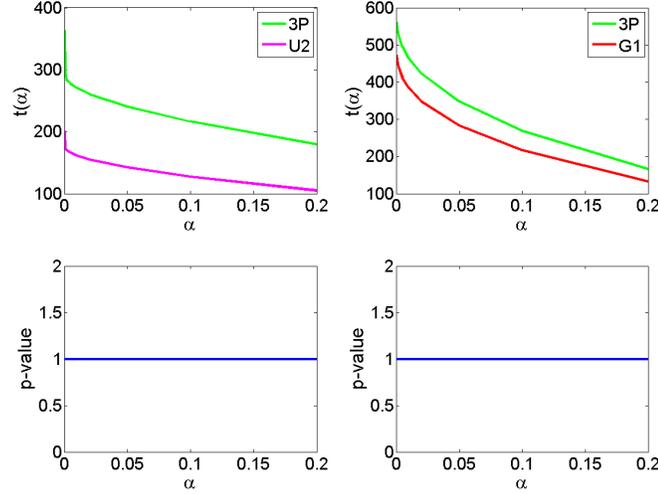


Figure 7.11: Comparison of $t(\alpha)$ for two different coefficient distributions specified in Table 7.2. Top-left: $t(\alpha)$ for FISTA, at $\delta = .3$, $\epsilon = .03$, $\lambda = .001$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Top-right: $t(\alpha)$ for IST, $\delta = .3$, $\epsilon = .07$, $\lambda = .05$, $\sigma = .001$. Bottom-right: The p-value of the test (7.20). Large p-values indicate that we cannot reject the null.

7.6.4 Discussions of the sparsity level

Proofs of the theorems

Lemma 7.6.5. *Define the risk of the soft thresholding function as,*

$$r_s(\mu, \tau; \sigma) = \mathbb{E}(\eta(\mu + \sigma Z; \tau\sigma) - \mu)^2;$$

We then have

$$\frac{d}{d\mu} r_s(\mu, \tau; \sigma) = 2\mu \int_{-\tau-\mu/\sigma}^{\tau+\mu/\sigma} \phi_Z(z) dz.$$

Proof.

$$\begin{aligned} \frac{d}{d\mu} r_s(\mu, \tau; \sigma) &= -2\mathbb{E}[(\eta(\mu + \sigma Z; \tau\sigma) - \mu)\mathbb{I}(|\mu + \sigma Z| \leq \tau\sigma)] \\ &= 2\mu\mathbb{E}(\mathbb{I}(|\mu + \sigma Z| \leq \tau\sigma)) = 2\mu\mathbb{P}(|\mu + \sigma Z| \leq \tau\sigma). \end{aligned}$$

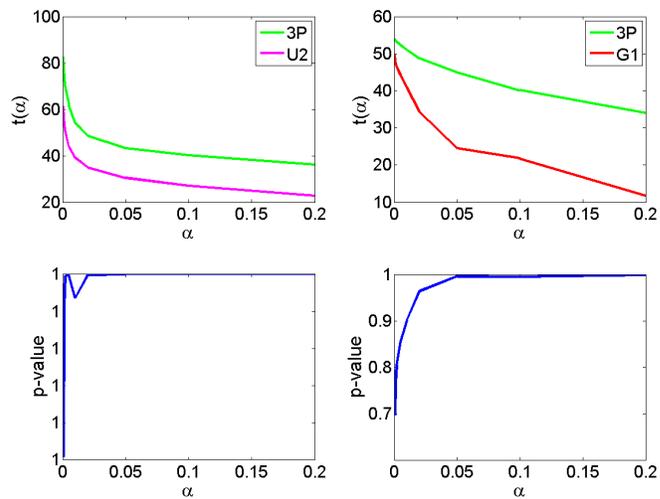


Figure 7.12: Comparison of $t(\alpha)$ for two different distributions on Top-left: $t(\alpha)$ for TWIST at $\delta = .3$, $\epsilon = .07$, $\lambda = .05$, $\sigma = .001$. Bottom-left: The p-value of the test (7.20). Top-right: $t(\alpha)$ of FPCAS at $\delta = .3$, $\epsilon = .07$, $\lambda = 0.01$, $\sigma = .001$. Bottom-right: The p-value of the test (7.20). Large p-values indicate that we cannot reject the null.

□

Lemma 7.6.6. *Consider the following risk function,*

$$R(\epsilon) = \mathbb{E}_{X \sim F_{\epsilon, 1/\sqrt{\epsilon}}} \mathbb{E}_X (\eta(X + \sigma Z; \tau\sigma) - X)^2;$$

$R(\epsilon)$ is an increasing function of ϵ .

Proof.

$$R(\epsilon) = (1 - \epsilon) \mathbb{E}(\eta(\sigma Z; \tau\sigma))^2 + \epsilon \mathbb{E}(\eta(\frac{1}{\sqrt{\epsilon}} + \sigma Z; \tau\sigma) - \frac{1}{\sqrt{\epsilon}})^2.$$

Therefore,

$$\begin{aligned} \frac{dR(\epsilon)}{d\epsilon} &= -\mathbb{E}(\eta(\sigma Z; \tau\sigma))^2 + \mathbb{E}(\eta(\frac{1}{\sqrt{\epsilon}} + \sigma Z; \tau\sigma) - \frac{1}{\sqrt{\epsilon}})^2 + \\ &\frac{2\epsilon}{2\epsilon^{1.5}} \mathbb{E}[(\eta(\frac{1}{\sqrt{\epsilon}} + \sigma Z; \tau\sigma) - \frac{1}{\sqrt{\epsilon}})(\mathbb{I}(|\frac{1}{\sqrt{\epsilon}} + \sigma Z| \leq \tau\sigma))] = \\ &-\mathbb{E}(\eta(\sigma Z; \tau\sigma))^2 + \mathbb{E}(\eta(\frac{1}{\sqrt{\epsilon}} + \sigma Z; \tau\sigma) - \frac{1}{\sqrt{\epsilon}})^2 - \frac{1}{\epsilon} P(|\frac{1}{\sqrt{\epsilon}} + \sigma Z| \leq \tau\sigma). \end{aligned} \quad (7.21)$$

We know that

$$\begin{aligned} &-\mathbb{E}(\eta(\sigma Z; \tau\sigma))^2 + \mathbb{E}(\eta(\frac{1}{\sqrt{\epsilon}} + \sigma Z; \tau\sigma) - \frac{1}{\sqrt{\epsilon}})^2 \\ &= \int_0^{\frac{1}{\sqrt{\epsilon}}} \frac{d}{d\mu} r_s(\mu, \tau; \sigma) d\mu = \int_0^{\frac{1}{\sqrt{\epsilon}}} 2\mu \int_{-\tau-\mu/\sigma}^{\tau+\mu/\sigma} \phi_Z(z) dz d\mu \\ &> \frac{1}{\epsilon} \int_{-\tau-1/(\sqrt{\epsilon}\sigma)}^{\tau+1/(\sqrt{\epsilon}\sigma)} \phi_Z(z) dz, \end{aligned} \quad (7.22)$$

where 1 is due to the fact that $\int_{-\tau-\mu/\sigma}^{\tau+\mu/\sigma} \phi_Z(z) dz$ is decreasing function of μ for $\mu > 0$ and therefore we replace μ with $\frac{1}{\sqrt{\epsilon}}$ we get something smaller. Combining (7.21) and (7.22) we conclude that,

$$\frac{dR(\epsilon)}{d\epsilon} > 0$$

which is what we intended to prove. □

Remark: It is clear that this theorem can be easily extended to more general types of distributions and the proof is essentially the same. Here is a generalization of the above theorem. Suppose that G is a distribution with $E_G(X^2) = 1$. We define $G_\epsilon(x) = (1 - \epsilon)\delta_0 + \epsilon G(x)$.

Lemma 7.6.7. *Consider the following risk function,*

$$R_G(\epsilon) = \mathbb{E}_{X \sim 1/\sqrt{\epsilon}G_\epsilon(x/\sqrt{\epsilon})} \mathbb{E}_X(\eta(X + \sigma Z; \tau\sigma) - X)^2;$$

$R_G(\epsilon)$ is an increasing function of ϵ .

As mentioned before since the proof is a simple modification of the previous proof we skip it.

Theorem 7.3.7 *If $\epsilon > \epsilon'$, $F_{\epsilon,1}$ is less favorable than $F_{\epsilon',1}$ for AMPT algorithm.*

Proof. According to Lemma 7.3.1, $F_{\epsilon,1}$ is equivalent to $F_{\epsilon,1/\sqrt{\epsilon}}$ and $F_{\epsilon',1}$ is equivalent to $F_{\epsilon',1/\sqrt{\epsilon'}}$. Therefore we compare these two distributions. These two new distributions have the same energy and therefore we do not need to scale the mean square error for comparison of these two distributions. We use mathematical induction to prove the above theorem. Let MSE_ϵ^t be the mean squared error of AMPT(τ) at iteration t on ϵ . Suppose that

$$MSE_\epsilon^t > MSE_{\epsilon'}^t.$$

Our goal is to prove,

$$MSE_\epsilon^{t+1} > MSE_{\epsilon'}^{t+1}.$$

We define

$$M\tilde{S}E_\epsilon^{t+1} = \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \mathbb{E}_X(\eta(X + \sqrt{\frac{MSE_{\epsilon'}^t}{\delta} + \sigma^2}Z; \tau\sqrt{\frac{MSE_{\epsilon'}^t}{\delta} + \sigma^2}) - X)^2;$$

We claim,

$$MSE_\epsilon^{t+1} > M\tilde{S}E_\epsilon^{t+1} > MSE_{\epsilon'}^{t+1}.$$

The first inequality is due to the fact that the mean square error is a non-decreasing

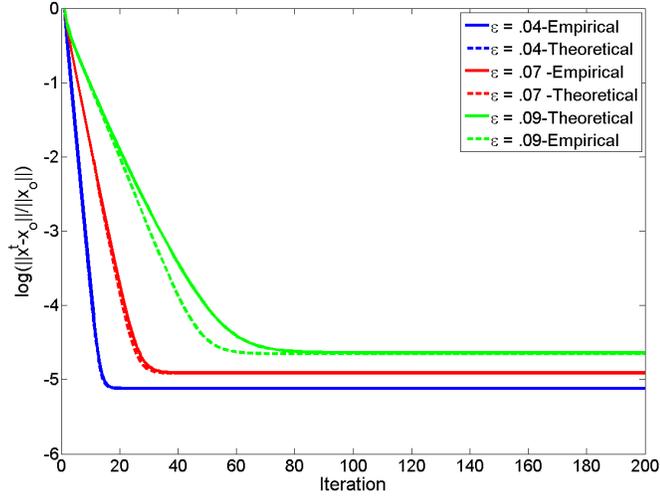


Figure 7.13: Comparison of different sparsity levels on AMPT algorithm $\delta = .5$, $\epsilon = .04$, $N = 4000$, $\tau = 1.8$, $\sqrt{\nu} = .001$, and the coefficient distribution is 3P. The empirical results are the result of 100 Monte Carlo samples.

function of the standard deviation of the noise. The second inequality however is the result of Lemma 7.6.6. The base of the induction is correct since both algorithms start with the same mean square error.

□

Figure 7.13 depicts the result of the above theorem on a problem instance.

Lemma 7.3.5 $F_{\epsilon,1}$ and $F_{\epsilon',1}$ are not equivalent for any of FISTA, IST and AMPT algorithms unless $\epsilon = \epsilon'$.

Proof. Suppose $\epsilon > \epsilon'$. Let us first prove that the two distributions are not equivalent for AMPT. Using Lemma 7.3.1 we consider the distributions $F_{\epsilon,1/\sqrt{\epsilon}}$ and $F_{\epsilon',1/\sqrt{\epsilon'}}$. The main advantage of the two new distributions is that their energy is equal to one and therefore relative mean square error is the same as the mean square error for them. According to the definition if these two distributions are equivalent for AMPT then for every τ_ϵ there exist $\tau_{\epsilon'}$ such that the mean square error of the two estimates are the same at every iteration. Choose $\tau_{\epsilon'}$ such that it achieves the minimum final mean

square error for $F_{\epsilon',1/\sqrt{\epsilon'}}$. Suppose there exists τ_ϵ achieves the same final mean square error. We now run AMPT($\tau_{\epsilon'}$) for distribution $F_{\epsilon',1/\sqrt{\epsilon'}}$ and according to Lemma 7.6.6 it will achieve an MSE which is less than the mean square error of τ_ϵ on $F_{\epsilon,1/\sqrt{\epsilon}}$ which is equal to the mean square error of $\tau_{\epsilon'}$ on $F_{\epsilon',1/\sqrt{\epsilon'}}$. This is a contradiction with our assumption that $\tau_{\epsilon'}$ is achieving the minimum mean square error for $F_{\epsilon',1/\sqrt{\epsilon'}}$. Therefore the two distributions are not equivalent for AMPT.

Clearly we can use a very similar argument for LASSO solvers. The only extra trick that we should use here is the equivalence of fixed point of LASSO and AMPT which was explained in Section 7.6.1. In other words we again start with the value of $\lambda_{\epsilon'}$ that achieves the minimum mean square error on $F_{\epsilon',1/\sqrt{\epsilon'}}$ and we prove that no λ_ϵ can achieve the same MSE. To prove this fact we use the relation between τ in AMPT and λ is LASSO(λ) and the fact that $F_{\epsilon,1/\sqrt{\epsilon}}$ is less favorable for AMPT. \square

Lemma 7.3.6 *Suppose that $\epsilon' \leq \epsilon$. For every $\lambda_\epsilon > 0$ and for any $\nu > 0$ if*

$$\lim_{N \rightarrow \infty} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2 \leq 1.$$

there exists a corresponding $\lambda_{\epsilon'}$ such that,

$$\lim_{N \rightarrow \infty} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2 = \lim_{N \rightarrow \infty} \mathbb{E}_{F_{\epsilon',1/\sqrt{\epsilon'}}} \|\hat{x}_{\lambda_{\epsilon'}} - x_o\|_2^2$$

Proof. With LASSO(λ_ϵ) we mean the problem where the coefficient is drawn from $F_{\epsilon,1/\sqrt{\epsilon}}$. According to Lemma 7.6.2 there exists a value τ_ϵ for which the asymptotic mean square error of the state evolution is the same as the asymptotic mean square error of the LASSO solver. If we consider $\underline{\tau} = \tau_\epsilon$, according to Lemma 7.6.6, the fixed point of AMPT($\underline{\tau}$) on $F_{\epsilon',1/\sqrt{\epsilon'}}$ is less than $\lim_{N \rightarrow \infty} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2$. If we set $\bar{\tau} = \infty$, The mean square error of AMPT($\bar{\tau}$) is larger than $\lim_{N \rightarrow \infty} \mathbb{E}_{F_{\epsilon,1/\sqrt{\epsilon}}} \|\hat{x}_{\lambda_\epsilon} - x_o\|_2^2$. Therefore there exists a value of $\tau_{\epsilon'}$ that gives us exactly the same means square error according to the implicit function theorem. If we use Lemma 7.6.2, we can conclude that there exists a value of $\lambda_{\epsilon'}$ with the same MSE on $F_{\epsilon',1/\sqrt{\epsilon'}}$. \square

Experimental setup for sparsity level experiments

Here we fixed the matrix distribution to $N(0, \frac{1}{n})$ and the coefficient distribution to $3P$ explained in the previous section. We also fixed the standard deviation of the noise $\sigma = .001$ and the relative mean square error to 1.6×10^{-4} . We considered 30 different equispaced values of δ and for every δ we also considered 30 different values of ρ from 0 to $.95\rho_{SE}(\delta)$. On each problem instance we set λ theoretically such that in the asymptotic regime $\mathbb{E}\|\hat{x}_\lambda - x_o\|_2^2 / \mathbb{E}\|x_o\|^2 = 1.6 \times 10^{-4}$. Figure 7.4 depicts the values of λ that result in this value of relative mean square error. These values are derived theoretically according to Lemma 7.3.6. In this approach we first find the values of τ that may lead to the given mean squared error in $AMPT(\tau)$. Then we calculate the corresponding values of λ for τ . Finally if there are more than one λ that lead to the same mean square error we consider the largest one (according to the observations of Section 7.4.2 gives the better convergence rate). Once the values of λ are calculated we set $N = 1500$ and we run the algorithms 20 times on each given grid point on the (ρ, δ) grid and take the average of the results. Some of the results are depicted in Figure 7.5.

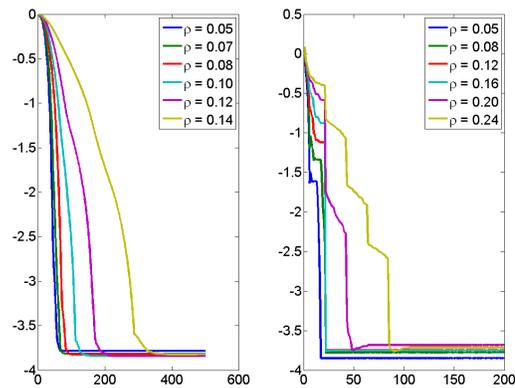


Figure 7.14: Comparison of different values of ρ or ϵ for TWIST and FPCAS. Left: $\delta = .07$, $N = 1500$, $\sigma = .001$, Algorithm = TWIST, matrix ensemble = $N(0, 1/n)$ and Coefficient ensemble is $F_{\epsilon,1}$. Right: $\delta = .48$, $N = 1500$, $\sigma = .001$, Algorithm = FPC-AS, matrix ensemble = $N(0, 1/n)$ and Coefficient ensemble is $F_{\epsilon,1}$. The sudden drops in FPC-AS correspond to the steps in which is solving a linear system.

Table 7.4: Performance of different algorithms on several compressed sensing problem instances. Here matrix ensemble is $N(0, 1/n)$ and the coefficient ensemble is 3P. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.

Algorithm	N	δ	ρ	τ or λ	ℓ_0 (sec)	ℓ_1 (sec)	ℓ_2 (sec)	ℓ_3 (sec)	fpe
AMP	2048	0.50	0.19	0.8769	0.001	0.007	0.016	0.026	1.187e-016
FISTA	2048	0.50	0.19	0.0220	0.014	0.038	NA	NA	4.093e-003
IST	2048	0.50	0.19	0.0220	0.051	0.235	NA	NA	4.093e-003
TWIST	2048	0.50	0.19	0.0220	0.010	0.023	NA	NA	4.093e-003
GPSR	2048	0.50	0.19	0.0220	0.016	0.065	NA	NA	4.093e-003
FPC	2048	0.50	0.19	0.0221	0.029	0.106	NA	NA	4.093e-003
FPC-AS	2048	0.50	0.19	0.0215	0.006	0.014	NA	NA	3.971e-003
AMP	2048	0.50	0.19	0.8769	0.001	0.007	0.016	0.026	1.187e-016
FISTA	2048	0.50	0.19	0.0022	0.038	0.123	0.168	NA	4.117e-004
IST	2048	0.50	0.19	0.0022	0.209	2.012	NA	NA	4.117e-004
TWIST	2048	0.50	0.19	0.0022	0.037	0.195	0.234	NA	4.117e-004
GPSR	2048	0.50	0.19	0.0022	0.053	0.596	0.762	NA	4.117e-004
FPC	2048	0.50	0.19	0.0022	0.028	0.092	0.179	NA	4.117e-004
FPC-AS	2048	0.50	0.19	0.0022	0.013	0.014	0.018	NA	4.202e-004
AMP	2048	0.50	0.19	0.8769	0.001	0.007	0.016	0.026	1.187e-016
FISTA	2048	0.50	0.19	0.0002	0.000	0.401	0.563	0.669	4.099e-005
IST	2048	0.50	0.19	0.0002	1.207	NA	NA	NA	4.099e-005
TWIST	2048	0.50	0.19	0.0002	0.472	NA	NA	NA	4.099e-005
GPSR	2048	0.50	0.19	0.0002	0.957	NA	NA	NA	4.099e-005
FPC	2048	0.50	0.19	0.0002	0.018	0.075	0.149	0.229	4.099e-005
FPC-AS	2048	0.50	0.19	0.0002	0.013	0.014	0.017	0.030	4.047e-005
AMP	1024	0.30	0.22	1.1924	0.000	0.002	0.005	0.008	1.702e-016
FISTA	1024	0.30	0.22	0.0002	0.057	0.126	0.174	.203	7.615e-005
IST	1024	0.30	0.22	0.0002	0.192	NA	NA	NA	7.615e-005
TWIST	1024	0.30	0.22	0.0002	0.280	NA	NA	NA	7.615e-005
GPSR	1024	0.30	0.22	0.0002	0.258	NA	NA	NA	7.615e-005
FPC	1024	0.30	0.22	0.0002	0.008	NA	NA	NA	7.615e-005
FPC-AS	1024	0.30	0.22	0.0002	0.002	0.004	0.005	0.007	7.307e-005
AMP	2048	0.30	0.22	1.1924	0.002	0.010	0.027	0.153	2.298e-011
FISTA	2048	0.30	0.22	0.0002	0.245	0.536	0.740	0.972	5.506e-005
IST	2048	0.30	0.22	0.0002	1.241	NA	NA	NA	5.506e-005
TWIST	2048	0.30	0.22	0.0002	1.444	NA	NA	NA	5.506e-005
GPSR	2048	0.30	0.22	0.0002	1.255	NA	NA	NA	5.506e-005
FPC	2048	0.30	0.22	0.0002	0.044	0.345	NA	NA	5.506e-005
FPC-AS	2048	0.30	0.22	0.0002	0.008	0.019	0.028	NA	2.190e-004

Table 7.5: Performance of different algorithms on several compressed sensing problem instances. Here the matrix ensemble is $N(0, 1/n)$ and the coefficient ensemble is Cauchy. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.

Algorithm	N	δ	ρ	τ or λ	l_0 (sec)	l_1 (sec)	l_2 (sec)	l_3 (sec)	fpe
AMP	2048	0.30	0.14	1.1924	0.017	0.003	0.005	0.008	3.453e-016
FISTA	2048	0.30	0.14	0.2837	NA	0.605	0.834	NA	3.622e-004
IST	2048	0.30	0.14	0.2837	NA	NA	NA	NA	3.622e-004
TWIST	2048	0.30	0.14	0.1917	NA	NA	NA	NA	3.622e-004
GPSR	2048	0.30	0.14	0.1917	NA	NA	NA	NA	3.622e-004
FPC	2048	0.30	0.14	0.2270	NA	0.016	0.040	NA	3.622e-004
FPC-AS	2048	0.30	0.14	0.1917	NA	0.004	0.009	NA	5.693e-004
AMP	2048	0.30	0.14	1.1924	0.017	0.003	0.005	0.008	3.453e-016
FISTA	2048	0.30	0.14	0.0284	NA	NA	NA	NA	2.370e-005
IST	2048	0.30	0.14	0.0284	NA	NA	NA	NA	2.545e-005
TWIST	2048	0.30	0.14	0.0171	NA	NA	NA	NA	2.545e-005
GPSR	2048	0.30	0.14	0.0171	NA	NA	NA	NA	2.545e-005
FPC	2048	0.30	0.14	0.0187	NA	0.015	0.023	.041	2.545e-005
FPC-AS	2048	0.30	0.14	0.0171	1.321	0.003	0.007	.017	2.545e-005

Table 7.6: Performance of different algorithms on several compressed sensing problem instances. Here the matrix ensemble is partial-DCT and the coefficient ensemble is $3P$. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.

Algorithm	N	δ	ρ	τ or λ	l_0 (sec)	l_1 (sec)	l_2 (sec)	l_3 (sec)	fpe
AMP	8192	0.50	0.19	0.8769	0.000	0.002	0.006	0.011	2.354e-017
FISTA	8192	0.50	0.19	0.0200	0.006	0.014	NA	NA	1.455e-003
IST	8192	0.50	0.19	0.0200	0.021	0.078	NA	NA	1.455e-003
TWIST	8192	0.50	0.19	0.0200	0.000	0.006	NA	NA	1.455e-003
GPSR	8192	0.50	0.19	0.0200	0.002	0.012	NA	NA	1.455e-003
FPC	8192	0.50	0.19	0.0198	0.003	0.006	NA	NA	1.455e-003
FPCAS	8192	0.50	0.19	0.0200	0.000	0.006	NA	NA	1.455e-003
AMP	8192	0.50	0.19	0.8769	0.000	0.002	0.006	0.011	2.354e-017
FISTA	8192	0.50	0.19	0.0019	0.017	0.051	0.072	NA	1.416e-004
IST	8192	0.50	0.19	0.0019	0.141	0.806	0.855	NA	1.416e-004
TWIST	8192	0.50	0.19	0.0019	0.002	0.025	0.036	NA	1.416e-004
GPSR	8192	0.50	0.19	0.0019	0.014	0.081	0.133	NA	1.416e-004
FPC	8192	0.50	0.19	0.0020	0.002	0.006	0.013	NA	1.416e-004
FPCAS	8192	0.50	0.19	0.0019	0.000	0.006	0.008	NA	1.416e-004

Table 7.7: Performance of different algorithms on several compressed sensing problem instances in the presence of measurement noise. Here the matrix ensemble is $N(0, 1/n)$ and coefficient ensemble is 3P. When the timing is shown by NA it means that the specified criteria has not been achieved by 3000 iterations of the algorithm.

Algorithm	N	δ	ρ	σ	ℓ_0 (sec)	ℓ_1 (sec)	ℓ_2 (sec)	fpe
AMP	2048	0.25	0.13	0.010	0.000	0.000	0.002	8.766e-003
FISTA	2048	0.25	0.13	0.010	0.002	0.004	NA	1.290e-002
IST	2048	0.25	0.13	0.010	0.005	0.017	NA	1.290e-002
TWIST	2048	0.25	0.13	0.010	0.001	0.002	NA	1.290e-002
GPSR	2048	0.25	0.13	0.010	0.001	0.004	NA	1.290e-002
FPC	2048	0.25	0.13	0.010	0.006	0.006	NA	1.290e-002
FPCAS	2048	0.25	0.13	0.010	0.002	0.002	NA	1.290e-002
AMP	2048	0.25	0.13	0.001	0.000	0.000	0.001	8.790e-004
FISTA	2048	0.25	0.13	0.001	0.000	0.012	0.020	1.279e-003
IST	2048	0.25	0.13	0.001	0.000	0.158	0.268	1.279e-003
TWIST	2048	0.25	0.13	0.001	0.000	0.010	0.015	1.279e-003
GPSR	2048	0.25	0.13	0.001	0.000	0.038	0.080	1.279e-003
FPC	2048	0.25	0.13	0.001	0.005	0.005	0.018	1.279e-003
FPCAS	2048	0.25	0.13	0.001	0.001	0.001	0.002	1.279e-003
AMP	2048	0.25	0.20	0.010	0.001	0.001	NA	1.208e-002
FISTA	2048	0.25	0.20	0.010	0.005	0.006	NA	2.138e-002
IST	2048	0.25	0.20	0.010	0.029	0.048	NA	2.138e-002
TWIST	2048	0.25	0.20	0.010	0.003	0.004	NA	2.138e-002
GPSR	2048	0.25	0.20	0.010	0.007	0.010	Na	2.138e-002
FPC	2048	0.25	0.20	0.010	0.013	0.012	NA	2.138e-002
FPCAS	2048	0.25	0.20	0.010	0.043	0.039	NA	2.138e-002
AMP	2048	0.25	0.20	0.001	0.001	0.001	0.004	1.398e-003
FISTA	2048	0.25	0.20	0.001	0.016	0.020	0.038	2.008e-003
IST	2048	0.25	0.20	0.001	0.268	0.269	0.269	2.008e-003
TWIST	2048	0.25	0.20	0.001	0.011	0.015	0.030	2.008e-003
GPSR	2048	0.25	0.20	0.001	0.073	0.088	0.271	2.008e-003
FPC	2048	0.25	0.20	0.001	0.017	0.011	0.084	2.008e-003
FPCAS	2048	0.25	0.20	0.001	0.002	0.002	0.008	2.008e-003

Chapter 8

Future Directions

In this chapter we explain some of the open problems and the directions that can be further explored.

8.1 Other sparsity patterns

In the first five chapters of this thesis we considered the ideal problem where the signals were exactly sparse and the measurements were accurate and there was no measurement noise. In chapter 6, we studied a more practical problem where the measurements were noisy and analyzed the asymptotic performance of LASSO and AMPT. However often in practice, natural signals are not exactly sparse and instead they have a few large coefficients and many small ones. This type of sparsity can be modeled in several different ways. One of the most popular ways to model this is to use ℓ_p balls for $0 \leq p \leq 1$ as explained in chapter 1. In these cases even when there is no measurement noise in the system, perfect recovery does not happen. Upper bounds have been derived in this setting [31], [25]. For instance, the following theorem due to Candés, Romberg and Tao [20] is one sample of these types of results.

Theorem 8.1.1. *Suppose the measurement matrix A satisfies the $RIP(2k, \sqrt{2} - 1)$. If \hat{x}_{ℓ_1} is the solution of basis pursuit problem and \hat{x}_o is the best k -term approximation*

of the original vector x in some ℓ_p -norm, then the following statement is true.

$$\|\hat{x}_{\ell_1} - \hat{x}_o\|_2 \leq C \frac{\|x - \hat{x}_o\|_1}{\sqrt{k}}.$$

for some constant C which is independent of the dimension of the problem.

From the above theorem the following bound can be derived for the performance of basis pursuit on ℓ_p balls.

Theorem 8.1.1. *Suppose that $x \in B_p^N(R)$ i.e. x is in the ℓ_p ball of radius R for $p \leq 1$. We take n random measurements with a Gaussian random matrix and use basis pursuit to reconstruct \hat{x}_{ℓ_1} . This reconstruction satisfies,*

$$\|\hat{x}_{\ell_1} - x\|_2 \leq C' \left(\frac{n}{\log(N/n)} \right)^{-1/p-1/2}.$$

Proof. We know from chapter 2 that if $n \sim k \log(N/k)$ then the matrix satisfies RIP of order k with overwhelmingly high probability. We prove that $\frac{n}{\log(N/n)} \leq \beta k$. clearly, $\frac{n}{\log(N/n)} \sim \frac{k \log(N/k)}{\log(N/k) - \log \log(N/k)}$. It is easy to show that $\log \log(x) < \frac{1}{2} \log(x)$ for every $x > 1$; Therefore, $k \lesssim \frac{n}{\log(N/n)}$.

According to Theorem 8.1.1,

$$\|\hat{x}_{\ell_1} - \hat{x}_o\|_2 \leq C \frac{\|x - \hat{x}_o\|_1}{\sqrt{k}} \leq \sum_{j=k+1}^{\infty} \frac{|x|_{(j)}}{\sqrt{k}} \leq \frac{CR}{k^{-1/p-1/2}} \leq C' \left(\frac{n}{\log(N/n)} \right)^{-1/p-1/2}.$$

where $|x|_{(j)}$ is the j^{th} largest element of the vector x . □

This result is due to [69],[61] for $p = 1$ and [31] for $p < 1$. It has been proved that the bound $\left(\frac{n}{\log(N/n)} \right)^{-1/p-1/2}$ is up to the constant optimal for both $p = 1$ [61] and $p < 1$ [31]. However, as before the constants are very loose and are not useful for practical purposes. State evolution may again provide a way to calculate the constants accurately in the asymptotic setting. The discussion of these questions is left for the future studies.

8.2 Correctness of state evolution

In chapter 4 we introduced the state evolution equation for the AMP algorithm. The claim was that in the asymptotic setting where $N \rightarrow \infty$, $\delta = n/N$ and $\rho = k/n$ these equations predict many different observables including false alarm rate, mean square error, prediction error, and missed detection rate of the algorithm at every iteration. Lots of simulation evidences were presented that confirmed the accuracy of these predictions. However the main questions that are left unanswered are 1) The theoretical validation of these equations and 2) The range of problem suites for which these predictions are accurate. Recently the correctness of the state evolution equation was proved for Gaussian measurement matrix. We quote the following theorem from [4]. Let $\{A(N)\}$ be a sequence of sensing matrices $A \in \mathbb{R}^{n \times N}$ indexed by N , with iid entries $A_{ij} \sim N(0, 1/n)$, and assume $n/N \rightarrow \delta$. Consider further a sequence of signals $\{x_0(N)\}_{N \geq 0}$, whose empirical distributions converge to a probability measure p_{X_0} on \mathbb{R} with bounded $(2k - 2)^{\text{th}}$ moment, and assume $\mathbb{E}_{\hat{p}}(X_0^{2k-2}) \rightarrow \mathbb{E}_{p_{X_0}}(X_0^{2k-2})$ as $N \rightarrow \infty$ for some $k \geq 2$.

Theorem 8.2.1. *For any pseudo-Lipschitz function $\psi : \mathbb{R}^2 \rightarrow \mathbb{R}$ we have,*

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \psi(x_i^{t+1}, x_{0,i}) = \mathbb{E}[\psi(\eta_t(X_0 + \tau_t Z), X_0)],$$

with $X_0 \sim p_{X_0}$ and $Z \sim N(0, 1)$ independent.

Unfortunately the proofs mentioned here can not be easily extended to other random matrices and does not explain the universality that has been observed in the simulation results. Therefore finding a more general framework for answering these questions and explaining the observed universality is an open question.

8.3 Convergence rate of first-order methods

In Chapter 7 we mentioned the first-order methods for solving the Lasso or basis pursuit denoising problems,

$$\min \|y - Ax\|_2^2 + \lambda\|x\|_1. \quad (8.1)$$

Let $f(x) = 1/2\|y - Ax\|_2^2 + \lambda\|x\|_1$. The following theorem is quoted again from that chapter,

Theorem 8.3.1. *For any $t \leq \frac{n-1}{2}$ and any λ, n, N , there exists a matrix $A \in \mathbb{R}^{n \times N}$ and a vector $y \in \mathbb{R}^n$ such that $|X^*| = 1$ and for any first-order method that uses the value of x^1, x^2, \dots, x^t and gradient of $\|y - Ax\|_2^2$ at x^1, \dots, x^t we have:*

$$\begin{aligned} f(x^t) - f(x^*) &\geq CL \frac{\|x_0 - x^*\|^2}{(t+1)^2}, \\ \|x^t - x^*\|_2^2 &\geq \frac{1}{25} \|x_0 - x^*\|_2^2. \end{aligned} \quad (8.2)$$

As mentioned before, this result is somewhat disappointing since it claims that the convergence of $x^t \rightarrow x^*$ can be in fact very slow. However it is based on the worst case analysis of the algorithm. A better notion of convergence rate for compressed sensing problems is the notion of average convergence rate since we are dealing with a random phenomena. Except for the AMP algorithm the average performance of these algorithms have not been analyzed theoretically yet. In Chapter 7, we characterized some of the characteristics of the convergence rates through an experimental studies. For instance, our experimental results confirmed the universality of the convergence rate with respect to the matrix ensemble. Furthermore, among the coefficient ensembles we tested the $3P$ or constant amplitude ensemble was the least favorable distribution for mean square error. However accurate theoretical predictions of the convergence rate will be more insightful. This question remains open for future studies.

Appendix A

Sub-gaussian and Sub-exponential Random Variables

In this section we summarize some of the properties of sub-gaussian and sub-exponential random variables used in the thesis. For better understanding of the results they may be compared with similar results for Gaussian and exponential random variables. The results are mainly taken from [108]. For more elaborate discussion of these topics the reader is referred to [108].

Lemma A.0.2. *Let X be a zero-mean random variable. The following statements are all equivalent.*

1. $\mathbb{P}(|X| > t) \leq C_1 e^{-c_1 t^2}$.
2. $\mathbb{E}(|X|^k)^{1/k} \leq C_2 \sqrt{k} \quad \forall k \geq 1$.
3. $\mathbb{E}(e^{cX^2}) \leq C_3$ for $0 \leq c \leq c_3$.
4. $\mathbb{E}(e^{tX}) \leq e^{ct^2} \quad \forall t \in \mathbb{R}$.

Proof. We will prove that $1 \Rightarrow 2$, $2 \Rightarrow 3$, $2, 3 \Rightarrow 4$, $3 \Rightarrow 1$ and finally $4 \Rightarrow 1$.

1 \Rightarrow 2:

$$\begin{aligned}\mathbb{E}(|X|^k) &= \int_0^\infty \mathbb{P}(|X|^k \geq t) dt \leq \int_{t=0}^\infty C_1 e^{-c_1 t^{2/k}} dt \\ &= \int_{t=0}^\infty C_1 k/2 z^{k/2-1} e^{-c_1 z} dz = C_1 k/2 \Gamma(k/2)/c_1^{k/2} \leq C_2^k \sqrt{k}^k.\end{aligned}$$

The last equality is derived from the normalization constant of the Gamma distribution.

2 \Rightarrow 3:

$$\mathbb{E}e^{cX^2} = \mathbb{E} \sum_{k=0}^{\infty} \frac{(cX^2)^k}{k!} \leq 1 + \sum_{k=1}^{\infty} \frac{c^k C_2^{2k} 2^k k^k}{k!} \leq 1 + \sum_{k=1}^{\infty} \frac{c^k C_2^{2k} 2^k k^k}{c' \sqrt{k} (k/e)^k}. \quad (\text{A.1})$$

Inequality (1) is due to the fact that $k! \geq c' \sqrt{k} \left(\frac{k}{e}\right)^k$. If $2ecC_2^2 < 1$ the last summation converges QED.

2, 3 \Rightarrow 4:

This is the only part for which X has to be zero mean. Since the random variable is zero mean, we can write the following.

$$\mathbb{E}(e^{tX}) \leq 1 + \sum_{k=2}^{\infty} t^k \frac{\mathbb{E}(X^k)}{k!} \leq 1 + \sum_{k=2}^{\infty} \frac{t^k C_2^k \sqrt{k}^k}{k!} \leq 1 + \sum_{k=2}^{\infty} \frac{t^k (C_2')^k}{\sqrt{k}^k}.$$

When $t < \frac{1}{C_2'}$ the right hand side is less than a geometric series. Therefore it is less than $1 + C_3' t^2/2 \leq e^{C_3' t^2}$. For the range $t < \frac{1}{C_2'}$ the proof is complete. Next step is to extend it to larger values of t . For this purpose we use part 3 of the lemma. Since $\mathbb{E}(e^{cX^2}) \leq C_3$, if $tX - ct^2 \leq c_2 X^2$ then we have $\mathbb{E}(e^{tX}) \leq C e^{ct^2}$ and since we just want to consider the values of $t > 1/C_2'$ we can just change the rate of exponential and have $\mathbb{E}(e^{tX}) \leq e^{ct^2}$ that we were looking for. Also $tX - ct^2 \leq c_2 X^2$ is clearly true for $c_2 = \frac{1}{4c}$.

3 \Rightarrow 1:

$$\mathbb{P}(|X| > u) = \mathbb{P}(\exp(c_2 X^2) > \exp(c_2 u^2)) \leq C_2 e^{-c_2 u^2},$$

where the last inequality is the result of the Markov inequality.

4 \Rightarrow 1:

$$\mathbb{P}(|X| > u) = \mathbb{P}(\exp(tX) > \exp(tu)) \leq e^{ct^2 - tu}.$$

If we do the optimization over the value of t we get the sub-gaussian tail. \square

Theorem A.0.3. *Let X_1, \dots, X_n be independent mean zero sub-gaussian random variables. A linear combination of these random variables is also a sub-gaussian random variable.*

Proof. Suppose that $Z = \sum_{i=1}^n a_i X_i$. Since these random variables are all zero mean Z is also zero mean and therefore according to the previous lemma it is sub-gaussian iff $\mathbb{E}(e^{t(\sum_{i=1}^n a_i X_i)}) \leq e^{ct^2}$. However since they are all independent calculating the expected value is simple,

$$\mathbb{E}(e^{t(\sum_{i=1}^n a_i X_i)}) \leq e^{t^2(\sum_{i=1}^n c_i a_i^2)},$$

and therefore this random variable is also sub-gaussian. \square

Definition A.0.1. *X is a sub-exponential random variable iff $\mathbb{P}(|X| > t) \leq Ce^{-ct}$, $\forall t > 0$.*

Lemma A.0.4. *Let X be a zero mean sub-exponential random variable. Then the following statements are true.*

1. $\mathbb{E}(|X|^p)^{1/p} \leq Cp$ for $p = 1, 2, 3, \dots$
2. $\mathbb{E}(e^{\lambda X}) \leq e^{C_1 \lambda^2}$ for $\lambda < C_1$.

Since the proof is very similar to the proof of Lemma A.0.2 it is skipped here. The following theorem from [108] shows the concentration of measure for the linear combination of sub-exponential random variables.

Theorem A.0.5. *Let X_1, X_2, \dots, X_n be iid sub-exponential random variables with $\mathbb{E}x_i = 0$, $a_1, a_2, \dots, a_n \in \mathbb{R}$. Let $T = \frac{\|a\|_2}{\|a\|_\infty}$. Then for all $t > 0$,*

$$\mathbb{P}\left(\left|\sum_k a_k x_k\right| > t\|a\|_2\right) \leq \begin{cases} ce^{-ct^2} & \text{if } t \leq T, \\ ce^{-cTt} & \text{otherwise.} \end{cases}$$

The main idea in the above proof is to use Markov inequality for $e^{\sum_k a_k x_k}$ and optimize over λ . This approach was done several times in Chapter 2 and we do not repeat it here. For the details, the reader may refer to [108].

Appendix B

Berry-Esseen Central Limit Theorem

In this section we present a proof of the Berry-Esseen central limit theorem which is used in Section 5.3 and Section 5.2. This proof is mainly due to Charles Stein [98]. Although more widely known approaches such as Lindeberg swapping trick [8] can also be used for proving similar results, we use Stein's method that gives stronger bound.

Theorem B.0.6. *Let S_1, S_2, \dots, S_n be independent zero mean random variables. Suppose $\mathbb{E}(S_i^2) = 1$ and $\mathbb{E}(|S_i|^3) \leq C$ where C is independent of i . For any bounded differentiable function $\phi(x)$ with bounded first derivative we have,*

$$\mathbb{E}(\phi(\frac{S_1 + S_2 + \dots + S_n}{\sqrt{n}})) = \mathbb{E}(\phi(G)) + O(\frac{C}{\sqrt{n}}(1 + \sup |\phi'(x)|)),$$

where $G \sim \mathcal{N}(0, 1)$.

Proof. Let $Z_n = \frac{S_1 + S_2 + \dots + S_n}{\sqrt{n}}$. Following Stein's method, for a given function $\phi(x)$ we define its Stein transform as,

$$T_\phi(x) = e^{\frac{x^2}{2}} \int_{-\infty}^x e^{-\frac{y^2}{2}} (\phi(y) - \mathbb{E}\phi(G)) dy.$$

This function is bounded and has bounded derivative and second order derivative and $\sup_x |T_\phi''(x)| \leq \sup_x |\phi'(x)|$. It is also not difficult to see that $\mathbb{E}(T_\phi'(Z_n) - Z_n T_\phi(Z_n)) = E(\phi(Z_n) - \phi(G))$. Define $T_i = Z_n - \frac{S_i}{\sqrt{n}}$.

$$\mathbb{E} Z_n T_\phi(Z_n) = \sum_i S_i T_\phi(Z_n) = \mathbb{E} \frac{1}{\sqrt{n}} \sum_i S_i T_\phi(T_i) + \frac{S_i^2}{\sqrt{n}} T_\phi'(T_i + t \frac{S_i}{\sqrt{n}}) = \frac{1}{n} \mathbb{E} S_i^2 T_\phi'(T_i + t \frac{S_i}{\sqrt{n}}).$$

We now try to bound $E(f'(Z_n) - Z_n f(Z_n))$.

$$\begin{aligned} \mathbb{E} (T_\phi'(Z_n) - Z_n T_\phi(Z_n)) &= \mathbb{E} T_\phi'(Z_n) + \frac{1}{n} \sum_i T_\phi'(T_i + \frac{S_i}{\sqrt{n}}) - S_i^2 T_\phi'(T_i + t \frac{S_i}{\sqrt{n}}) \\ &= \mathbb{E} \frac{1}{n} \sum_i T_\phi'(T_i + \frac{S_i}{\sqrt{n}}) - T_\phi'(T_i) + f'(T_i) - S_i^2 T_\phi'(T_i + t \frac{S_i}{\sqrt{n}}) \\ &= \mathbb{E} \frac{1}{n} \sum_i T_\phi'(T_i + \frac{S_i}{\sqrt{n}}) - T_\phi'(T_i) + S_i^2 T_\phi'(T_i) - S_i^2 T_\phi'(T_i + t \frac{S_i}{\sqrt{n}}) \\ &\leq \frac{1}{\sqrt{n}} E(|S_i| \sup_x |T_\phi''(x)|) + E(|S_i|^3) \sup_x |T_\phi''(x)| \\ &\leq \frac{4}{\sqrt{n}} \sup |\phi'(x)|. \end{aligned}$$

□

Theorem B.0.7. *Let S_1, S_2, \dots, S_n be independent zero mean random variables. Suppose $\mathbb{E}(S_i^2) = 1$ and $\mathbb{E}(|S_i|^3) \leq C$ where C is independent of i .*

$$\mathbb{P}(Z_n < a) = \mathbb{P}(G < a) + O\left(\frac{1}{\sqrt{n}}\right)$$

uniformly for all $a \in \mathbb{R}$, where $G \sim N(0, 1)$.

Proof. The proof is very similar to what we did for the previous theorem. We define $\phi(x) = \mathbb{I}(x < a)$ and use the Stein's method. Although the function is not differentiable any more, it is possible to see that $xT_\phi(x)$ is Lipschitz and we can use similar arguments. □

Appendix C

Wigner's Semicircle Law

Theorem C.0.2. [111, 112] *Let A be a real symmetric $N \times N$ matrix. Suppose that for $i \geq j$, the elements A_{ij} are drawn iid. from a distribution F . Further, assume that $E_F(A_{ij}^2) = 1/N$. If S is the number of eigenvalues of A that lie in the interval (α, β) , for real $\alpha < \beta$, then,*

$$\lim_{N \rightarrow \infty} \mathbb{E}(S) = \frac{1}{2\pi} \int_{\alpha}^{\beta} \sqrt{4 - x^2} dx.$$

Suppose that the empirical distribution of the eigenvalues of a given matrix is called μ_{A_N} . This is of course a probability measure, but it may depend on the ensemble we draw and therefore it is random itself. The above theorem implies that the expected value of these distributions, which we may call μ_N , converges to the semicircle law.

Lemma C.0.8. *Let A be as given in the previous theorem. We have*

$$\mathbb{E} \frac{1}{N} \text{Tr}(A^k) = \int x^k d\mu_N(x).$$

Proof.

$$\frac{1}{N} \text{Tr}(A^k) = \int x^k d\mu_{A_N}(x).$$

Taking expectation from both sides will give us the result. \square

Theorem C.0.3. *Let A be as given in Theorem C.0.2. Furthermore, suppose F is sub-gaussian. If γ is an eigenvalue of A ,*

$$\lim_{N \rightarrow \infty} \int x^k d\mu_N(x) = \frac{1}{2\pi} \int_{-2}^2 \gamma^k \sqrt{4 - \gamma^2} d\gamma, \quad \forall k \in \mathbb{N}.$$

Proof. According to Wigner's theorem we know that $\mu_N(x)$ is converging to the semi-circle law. It is not difficult to prove that the maximum and minimum eigenvalues of this matrix are also sub-gaussian; We did it for similar cases in Chapter 2. Therefore according to Theorem 4, Lecture 1 of [101], we conclude the above theorem. \square

Appendix D

Proofs of Chapter 2

D.1 Proof of Theorem 2.3.1

The goal of this section is to give an outline of the proof of Theorem 2.3.1. **Theorem 2.3.1** Suppose that $k < \frac{1}{3.1}\mu^{-1}$ and $\frac{|x_o(i)|}{|x_o(i+1)|} < 3^{\ell_i-4}, \forall i, 1 \leq i < k$. Then IHT finds the correct active set in at most $\sum_{i=1}^k \ell_i + k$ steps. After this step all of these elements will remain in the active set and the error will go to zero exponentially fast.

Proof. Let me first summarize the behavior of the algorithm intuitively. It will help the reader understand the steps of the proof more easily. All these things will be proved rigorously later in this section. When we run the algorithm, at the first iteration the largest element of x_o will get into the active set. Interestingly, once this element gets into the active set it will always remain in the active set. In the next few iterations of the algorithm the first element remains in the active set and the error term decreases and finally it will be so small that the second largest element will be detected (This statement is not exactly right. The error term may go up for a finite number of iterations, but eventually it will decrease. You will see the rigorous bound of this error and its performance in the next lemma). Once the second largest term gets into the active set, the first and second elements will remain in the active set and the same process will happen again, i.e. the error term will decrease and eventually

the third largest element will get into the active set. The goal of this section is to make all the above statements precise.

The next lemma will be useful later when we try to bound the error at each iteration.

Lemma D.1.1. *Consider the following sequence for $s \geq 0$,*

$$f_s = \alpha^1 + \dots + \alpha^s + \beta\alpha^{s+1},$$

where $0 < \alpha < 1$. The following statements are true;

1. If $\beta(1 - \alpha) < 1$, then for every s , $f_s < \frac{\alpha}{1-\alpha}$.
2. If $\beta(1 - \alpha) > 1$, then for every s , $f_s < \beta\alpha$.
3. If $\beta(1 - \alpha) = 1$, then f_s is a constant sequence and is always equal to $\frac{\alpha}{1-\alpha}$.

It is easy to see that the sequence is either increasing or decreasing or constant depending on the values of α and β . The proof is simple and is omitted for the sake of brevity.

Lemma D.1.2. *Suppose that $x_o(1), x_o(2), \dots, x_o(r-1)$, $r-1 < k$, are in the active set at the m^{th} step. Also assume that,*

$$|z^m(j) - x_o(j)| \leq 1.5 \cdot k \cdot \mu |x_o(r-1)| \quad \forall j.$$

If $k\mu < \frac{1}{3.1}$, then at stage $m+s$ and for every j , we have the following upper bound for $|z^{m+s}(j) - x_o(j)|$:

$$|x_o(r)| (k\mu + \dots + (k\mu)^s) + 1.5(k\mu)^{s+1} |x_o(r-1)|. \quad (\text{D.1})$$

Moreover, $x_o(1), x_o(2), \dots, x_o(r-1)$ will remain in the active set.

Before proving this theorem, it should be mentioned that at this point, the factor 1.5 may seem unnecessary in the proof. But as will be seen later in Lemma D.1.3, this factor is necessary and can not be omitted.

Proof. We prove this by induction; Assuming that the bound holds at stage $m + s$ and $x_o(1), x_o(2), \dots, x_o(r - 1)$ are in the active set, we show that the upper bound holds at stage $m + s + 1$ and the first $r - 1$ elements will remain in the active set.

$$\begin{aligned}
& |z^{m+s+1}(i) - x_o(i)| \\
& \leq \left| \sum_{j=1}^k \langle A_i, A_j \rangle w_j^{m+\ell} + \sum_{j \in I^{m+\ell}} \langle A_i, A_j \rangle w_j^{m+\ell} \right|, \\
& \leq \sum_{j \in I^{m+s} \setminus \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)| + \sum_{j \in \{1,2,\dots,k\} \setminus I^{m+s} \cup \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)|, \\
& \stackrel{1}{=} \sum_{j \in I^{m+s} \setminus \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)| + \sum_{j \in \{r,\dots,k\} \setminus I^{m+s} \cup \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)|, \\
& \stackrel{2}{\leq} \sum_{j \in I^{m+s} \setminus \{i\}} |\langle A_i, A_j \rangle (z^{m+s}(j) - x_o(j))| + k\mu x_o(r), \\
& \leq k\mu |x_o(r)| (k\mu + \dots + (k\mu)^s) + 1.5(k\mu)^{s+2} |x_o(r - 1)| \\
& \quad + k\mu |x_o(r)|, \\
& \leq |x_o(r)| (k\mu + \dots + (k\mu)^{s+1}) + 1.5(k\mu)^{s+2} |x_o(r - 1)|.
\end{aligned}$$

In these calculations equality (1) is due to the assumptions of the induction, i.e. the first $r - 1$ elements are in the active set at stage $m + s$. To get inequality (2) we have used two different facts. The first one is that when $j \in I^{m+s}$, $w^{m+s}(j) = x_o(j) - z^{m+s}(j)$ and the second one is that when $j \in \{r, \dots, k\} \setminus I^{m+s}$ then $w^{m+s}(j) = x_o(j)$ and therefore $|x_o(j)| \leq |x_o(r)|$. The last step is to prove that all the first $r - 1$ elements remain in the active set. For $i \in \{1, 2, \dots, r - 1\}$,

$$\begin{aligned}
& |z^{m+s+1}(i)| \geq |x_o(i)| - |z^{m+s+1}(i) - x_o(i)|, \\
& \stackrel{1}{\geq} |x_o(i)| - (k\mu |x_o(r - 1)| + \dots + (k\mu)^{s+1} |x_o(r - 1)|) \\
& \quad - 1.5(k\mu)^{s+2} |x_o(r - 1)| \stackrel{2}{\geq} |x_o(i)| - \frac{|x_o(r - 1)|}{2.05} \\
& \geq |x_o(r - 1)| - \frac{|x_o(r - 1)|}{2.05}.
\end{aligned}$$

In inequality (1) we have used the bound in (D.1) by replacing $x_o(r)$ with $x_o(r-1)$. Inequality (2) is the result of Lemma D.1.1. For $i \notin \{1, 2, \dots, k\}$, we have

$$|z^{m+s+1}(i)| \leq \frac{|x_o(r-1)|}{2.05},$$

and since $\min_{\{i:i \leq r-1\}} |z^{m+s+1}(i)| > \max_{\{i:i > k\}} |z^{m+s+1}(i)|$, the first $r-1$ elements will remain in the active set. The base of the induction is the same as the assumptions of this lemma. \square

Lemma D.1.3. *Suppose that $k < \frac{1}{3.1}\mu^{-1}$, and $x_o(1), x_o(2), \dots, x_o(r)$, $r < k$, are in the active set at the m^{th} step. Also assume that $\frac{|x_o(r)|}{|x_o(r+1)|} \leq 3^{\ell_r-4}$. If*

$$|z^m(j) - x_o(j)| \leq 1.5 \cdot k \cdot \mu |x_o(r)| \quad \forall j,$$

after ℓ_r more steps $x_o(r+1)$ will get into the active set, and

$$|z^{m+\ell_r+1}(j) - x_o(j)| \leq 1.5 \cdot k \cdot \mu |x_o(r+1)| \quad \forall j.$$

Proof. By setting $s = \ell_r$ in the upper bound of the last lemma we get,

$$|z^{m+\ell_r}(j) - x_o(j)| \leq \frac{1.5|x_o(r+1)|}{273} + \frac{|x_o(r+1)|}{2.1}.$$

Similar to the last lemma it is also not difficult to see that

$$\begin{aligned} |z^{m+\ell_r}(r+1)| &= |z^{m+\ell_r}(r+1) - x_o(r+1) + x_o(r+1)| \\ &\geq |x_o(r+1)| - |z^{m+\ell_r}(r+1) - x_o(r+1)| \\ &\geq |x_o(r+1)| - \frac{|1.5x_o(r+1)|}{273} - \frac{|x_o(r+1)|}{2.1}. \end{aligned}$$

But,

$$|z^{m+\ell_r}(r+1)| > \max_{\{i:i > k\}} |z^{m+\ell_r}(i)|,$$

and therefore $x_o(r+1)$ will be detected at this step. It may also be noted that at this stage the error is less than $|x_o(r+1)|/2$. For the next stage we will have at most k

active elements the error of each is less than $|x_o(r+1)|/2$ and at most $k-r$ non-zero elements of x_o that have not passed the threshold and whose magnitudes are smaller than $|x_o(r+1)|$. Therefore, the error of the next step is less than $1.5k\mu|x_o(r+1)|$. \square

Our goal is to prove the correctness of IHT by induction and we have to know the correctness of IHT at the first stage. The following lemma provides this missing step.

Lemma D.1.4. *Suppose that $k < \frac{1}{3.1}\mu^{-1}$, then at the first stage of the IHT, $x_o(1)$ will be in the active set¹ and for every j , $|z^1(j) - x_o(j)| \leq k\mu|x_o(1)|$.*

The proof is very simple and is omitted.

Finally the following lemma describes the performance of the algorithm after detecting all the non-zero elements.

Lemma D.1.5. *Suppose that $x_o(1), x_o(2), \dots, x_o(k)$, are in the active set at the m^{th} step. Also assume that,*

$$|z^m(j) - x_o(j)| \leq 1.5 \cdot k \cdot \mu |x_o(k)| \quad \forall j.$$

If $k\mu < \frac{1}{3.1}$, then at stage $m+s$ and for every j , we have

$$|z^{m+s}(j) - x_o(j)| \leq 1.5(k\mu)^{s+1}|x_o(k)|.$$

Since the proof of this lemma is very simple, it is omitted.

The proof of the main theorem is an induction that combines the above lemmas. Suppose that $x_o(1), x_o(2), \dots, x_o(r)$ are already in the active set. According to Lemma D.1.2 all these terms will remain in the active set, and according to Lemma D.1.3 after ℓ_r steps $x_o(r+1)$ will also get into the active set. In one more step, the error on each element gets smaller than $1.5k\mu|x_o(r+1)|$, and everything can be repeated. Lemma D.1.4 provides the first step of the induction. Finally when all the elements are in the active set Lemma D.1.5 tells us that the error goes to zero exponentially fast.

\square

¹This result holds even if $k\mu < \frac{1}{2}$. For the sake of consistency with the other parts of the proof we state it in this way

D.2 Proof of Theorem 2.4.1

Theorem 2.4.1 Suppose that $k < \frac{1}{4.1}\mu^{-1}$ and $\forall i, 1 \leq i < k$, we have $\frac{|x_o(i)|}{|x_o(i+1)|} < 2^{\ell_i - 5}$. Then IST recovers the correct active set in at most $\sum_{i=1}^k \ell_i + k$ steps. After that all these coefficients will remain in the active set and the error will go to zero exponentially fast.

Proof. As mentioned before the main ideas of the proof of the IST algorithm are very similar to those of the IHT. We will mention the proof in detail but will try to emphasize more on the differences. The following lemma helps us find some bounds on the error of the algorithm at each step.

Lemma D.2.1. *Suppose that $x_o(1), x_o(2), \dots, x_o(r)$, $r \leq k$, are in the active set at the m^{th} step. Also assume that*

$$|x^m(j) - x_o(j)| \leq 4 \cdot k \cdot \mu |x_o(r)|, \quad \forall j \in I^m,$$

and $k\mu < \frac{1}{4.1}$. Then at stage $m+s$, $\forall i \in I^{m+s}$ we have the following upper bound for $|x^{m+s}(i) - x_o(i)|$,

$$|x_o(r+1)| (2k\mu + \dots + (2k\mu)^s) + 2(2k\mu)^{s+1} |x_o(r)|.$$

Moreover, $x_o(1), x_o(2), \dots, x_o(r)$ remain in the active set.

Proof. As before, this can be proved by induction. We assume that at step $m+s$ the upper bound holds and $x_o(1), x_o(2), \dots, x_o(r)$ are in the active set and we prove the

same things for $m + s + 1$. Similar to what we saw before,

$$\begin{aligned}
& |z^{m+s+1}(i) - x_o(i)| \\
& \leq \sum_{j \in I^{m+s} \setminus \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)| + \sum_{j \in \{1,2,\dots,k\} \setminus I^{m+s} \cup \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)|, \\
& \stackrel{1}{=} \sum_{j \in I^{m+s} \setminus \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)| + \sum_{j \in \{r+1,\dots,k\} \setminus I^{m+s} \cup \{i\}} |\langle A_i, A_j \rangle w^{m+s}(j)|, \\
& \stackrel{2}{\leq} (k-1)\mu(2k\mu|x_o(r+1)| + \dots + (2k\mu)^s|x_o(r+1)| \\
& + 2(2k\mu)^{s+1}|x_o(r)|) + k\mu|x_o(r+1)| := \alpha_s.
\end{aligned}$$

Equality (1) is using the assumption that the first r elements are in the active set at stage $m + s$. Inequality (2) is also due to the assumptions of the induction and the fact that $w^{m+s}(j) = x_o(j) - x^{m+s}(j)$.

At least one of the largest $k+1$ coefficients of z , corresponds to an element whose index is not in $\{1, 2, \dots, k\}$, and the magnitude of this coefficient is less than α_s . Therefore the threshold value is less than or equal to α_s . Applying the soft thresholding to z will at most add α_s to the distance of $z^{s+1}(i)$ and $x_o(i)$, and this completes the proof of the upper bound. The main thing that should be checked is whether the first r elements will remain in the active set or not. For $i \in \{1, 2 \dots r\}$ we have,

$$\begin{aligned}
|z^{m+s+1}(i)| & \geq |x_o(i)| - |z^{m+s+1}(i) - x_o(i)|, \\
& \geq |x_o(i)| - k\mu|x_o(r)|(1 + 2k\mu + \dots + (2k\mu)^{s+1}) \\
& \quad - 2k\mu(2k\mu)^{s+1}|x_o(r)| \geq |x_o(i)| - \frac{|x_o(r)|}{2.05} \\
& \geq |x_o(r)| - \frac{|x_o(r)|}{2.05}. \tag{D.2}
\end{aligned}$$

If the sequence in the above expression is multiplied by 2, the result will be a sequence in the form of the sequences mentioned in Lemma D.1.1 for $\alpha = 2k\mu$, $\beta = 2$ and the last equality is based on that lemma.

If $i \notin \{1, 2 \dots k\}$,

$$\begin{aligned} |z^{m+s+1}(i)| &\leq k\mu|x_o(r)|(1 + 2k\mu + \dots + (2k\mu)^{s+1}) \\ &\quad + 2k\mu(2k\mu)^{s+1}|x_o(r)| \leq \frac{|x_o(r)|}{2.05}. \end{aligned}$$

Since $\min_{\{i:i \leq r\}} |z^{m+s+1}(i)| > \max_{\{i:i > k\}} |z^{m+s+1}(i)|$, the first r elements remain in the active set. The base of the induction is also clear since it is the same as the assumptions of the lemma. \square

Lemma D.2.2. *Suppose that $k \leq \frac{\mu^{-1}}{4.1}$, and $x_o(1), x_o(2), \dots, x_o(r)$, $r \leq k$, are in the active set at the m^{th} step. Also, assume that $\frac{|x_o(r)|}{|x_o(r+1)|} \leq 2^{\ell_r - 5}$. If*

$$|x^m(j) - x_o(j)| \leq 4 \cdot k \cdot \mu |x_o(r)|, \quad \forall j \in I^m,$$

then after ℓ_r steps $x_o(r+1)$ will get into the active set, and

$$|x^{m+\ell_r+1}(j) - x_o(j)| \leq 4k\mu|x_o(r+1)|, \quad \forall j \in I^{m+\ell_r+1}.$$

Proof. As before, we try to find a bound for the error at time $m + \ell_r$. For $i \in \{1, 2, \dots, k\}$,

$$\begin{aligned} |z^{m+\ell_r}(i) - x_o(i)| &\leq \frac{1}{2}|x_o(r+1)|(2k\mu + \dots + (2k\mu)^{\ell_r}) \\ &\quad + (2k\mu)^{\ell_r+1}|x_o(r)| \leq \frac{|x_o(r+1)|}{2.1} + \frac{|x_o(r+1)|}{64} \end{aligned}$$

and therefore, for $i = r+1$,

$$\begin{aligned} |z^{m+\ell_r}(r+1)| &\geq |x_o(r+1)| - |z^{m+\ell_r}(i) - x_o(i)| \geq \\ &\quad |x_o(r+1)| - \frac{|x_o(r+1)|}{2.1} - \frac{|x_o(r+1)|}{64} \end{aligned} \tag{D.3}$$

Since $|z^{m+\ell_r}(r+1)| > \max_{\{i:k < i\}} |z^{m+\ell_r}(i)|$, the $r+1^{\text{th}}$ element will get into the active set at this stage. On the other hand for any $i \in I^{m+\ell_r}$ we have $|x^{m+\ell_r}(i) - x_o(i)| \leq$

$x_o(r+1)$. For the next stage of the algorithm we will have at most $2k$ non-zero $x^{m+\ell_r}(i) - x_o(i)$ and absolute value of each of them is less than $|x_o(r+1)|$. Therefore $|z^{m+\ell_r+1}(i) - x_o(i)| \leq 2k\mu|x_o(r+1)|$ and after thresholding we have, $|x^{m+\ell_r+1}(i) - x_o(i)| \leq 4k\mu|x_o(r+1)|$ for $i \in I^{m+\ell_r+1}$.

The base of the induction is also clear from the assumptions of this lemma. \square

For the IHT algorithm we proved that at the first step the first element will pass the threshold. Since the selection step of IST and IHT is exactly the same, we can claim that the same thing is true for IST, i.e. the largest magnitude coefficient will pass the threshold. Also, as we saw for IHT, the error was less than $k\mu|x_o(1)|$. Therefore, for the IST we have $|x^1(j) - x_o(j)| < 2k\mu|x_o(1)|$ for every j . These bounds are even better than the bounds we need for D.2.1 and D.2.2 and D.2.3.

The following lemma will explain what happens when the algorithm detects all the non-zero elements.

Lemma D.2.3. *Suppose that $x_o(1), \dots, x_o(k)$, are in the active set at the m^{th} step. Also assume that,*

$$|x^m(j) - x_o(j)| \leq 4 \cdot k \cdot \mu |x_o(k)|.$$

If $k\mu < \frac{1}{41}$, at stage $m+s$ all the elements remain in the active set and for every j we will have,

$$|z^{m+s}(j) - x_o(j)| \leq 2(2k\mu)^{s+1}|x_o(k)|$$

The proof of this lemma is very similar to the other lemmas and is omitted.

The proof of the main theorem is a simple induction by combining the above lemmas. Suppose that $x_o(1), x_o(2), \dots, x_o(r)$ are already in the active set. According to Lemma D.2.1 all these terms will remain in the active set, and according to Lemma D.2.2 after ℓ_r steps $x_o(r+1)$ will also get into the active set. In one more step, the error on each element gets smaller than $4k\mu|x_o(r+1)|$, and everything can be repeated. Although we have not mentioned the first step of the induction, it is not difficult to see that it is also true and it is very similar to the first step of IHT. Finally when all

the elements are in the active set, Lemma D.2.3 ensures that the error goes to zero exponentially fast. \square

Appendix E

Proofs of Chapter 4

E.1 proof of Lemma 4.2.1

Lemma 4.2.1 Let A_{ij} be the ij^{th} element of the measurement matrix and suppose that these elements are drawn iid from a given distribution with the following properties, $\mathbb{E}(A_{ij}) = 0$, $\mathbb{E}(A_{ij}^2) = \frac{1}{n}$ and $\mathbb{E}(A_{ij}^4) = O(\frac{1}{n^2})$. We assume that the ratio $\delta = \frac{n}{N}$ is fixed. Let the vector x be a vector with elements iid from a possibly different distribution, with a bounded second moment and independent of the matrix A . If $s = (A^*A - I)x$, then

$$\begin{aligned}\mathbb{E}(s_i) &= 0, \\ \mathbb{E}(s_i s_j) &= O\left(\frac{1}{n}\right), \\ \mathbb{E}(s_i^2) &= \frac{\mathbb{E}(x_i^2)}{\delta} + O\left(\frac{1}{n}\right).\end{aligned}$$

Proof. In this proof x_i represents the i^{th} element of x and $H = A^*A - I$. H_i and A_i the i^{th} column of the H and A matrices respectively and H_{ij} and A_{ij} are the ij^{th} elements of H and A . It is easy to see that,

$$\mathbb{E}(H_{ij}) = 0. \tag{E.1}$$

Therefore,

$$\mathbb{E}(s_i) = \mathbb{E}\left(\sum_j H_{ij}x_j\right) = \sum_j E(H_{ij})E(x_j) = 0.$$

It is also clear that H_{ik} depends on the i^{th} and k^{th} columns of the A matrix. Hence if $i \neq j$, $i \neq \ell$, $j \neq k$ and $j \neq \ell$,

$$\mathbb{E}(H_{ij}H_{k\ell}) = 0. \quad (\text{E.2})$$

Now consider another case $i \neq j$ and $\ell \neq i, j$.

$$\mathbb{E}(H_{ij}H_{j\ell}) = \mathbb{E}(A_i^*A_jA_j^*A_\ell) = \mathbb{E}(A_i^*)\mathbb{E}(A_jA_j^*A_\ell) = 0.$$

The final case we consider here is $i \neq j$.

$$\mathbb{E}(H_{ij}^2) = \mathbb{E}(A_i^*A_jA_j^*A_i) = \frac{1}{n}\mathbb{E}(A_i^*A_i) = \frac{1}{n}.$$

$$\begin{aligned} \mathbb{E}(s_i s_j) &= \mathbb{E}\left(\left(\sum_k H_{ik}x_k\right)\left(\sum_\ell H_{j\ell}x_\ell\right)\right) = \\ &= \sum_k \sum_\ell \mathbb{E}(H_{ik}H_{j\ell})\mathbb{E}(x_k x_\ell) \stackrel{1}{=} \mathbb{E}(H_{ij}H_{ji})\mathbb{E}(x_j x_i) = O\left(\frac{1}{n}\right). \end{aligned} \quad (\text{E.3})$$

Equality (1) is the result of what we proved before for the covariance of H_{ij} and $H_{k\ell}$.

$$\mathbb{E}(s_i^2) = \mathbb{E}(x^* H_i H_i^* x) = E(x^* \mathbb{E}(H_i^* H_i) x).$$

From what we proved before $\mathbb{E}(H_i^* H_i)$ is a diagonal matrix and therefore,

$$E(x^* \mathbb{E}(H_i^* H_i) x) = E(x_i^2) \sum_k E(H_{ik}^2). \quad (\text{E.4})$$

We also have,

$$\begin{aligned} \sum_{k \neq i} \mathbb{E}((A_i^* A_k)^2) &= \sum_{k \neq i} \mathbb{E} \text{Tr}(A_i^* A_k A_k^* A_i) = \\ \sum_{k \neq i} \mathbb{E} \frac{1}{n^2} \text{Tr}(I_n) &= \frac{N-1}{n} = \frac{1}{\delta} + O\left(\frac{1}{n}\right). \end{aligned}$$

where I_n is the $n \times n$ identity matrix and

$$\begin{aligned} \mathbb{E}((A_i^* A_i - 1))^2 &= \mathbb{E}((A_i^* A_i)^2 - 2(A_i^* A_i) + 1) = \\ \mathbb{E}((A_i^* A_i)^2) + -2 + 1 &= \mathbb{E}\left(\sum_j A_{ij}^2 \sum_k A_{ik}^2\right) - 1 = \\ \sum_k \sum_{j \neq k} \mathbb{E}(A_{ij}^2 A_{ik}^2) + \sum_k \mathbb{E}(A_{ik}^4) &= \\ \frac{n(n-1)}{n^2} + O\left(\frac{1}{n}\right) - 1 &= O\left(\frac{1}{n}\right). \end{aligned} \tag{E.5}$$

By combining the three above equations we have,

$$\mathbb{E}(s_i^2) = \frac{E(x_i^2)}{\delta} + O\left(\frac{1}{n}\right). \tag{E.6}$$

□

E.2 Proof of Proposition 4.2.2

Proposition 4.2.2. For the three canonical problems $\chi \in \{+, \pm, \square\}$, any $\delta \in [0, 1]$, and any random variable X with the prescribed sparsity and bounded second moment, $\rho_{\text{SE}}(\delta; \chi, \lambda, F_X)$ is independent of F_X .

Since the calculations that are needed for each case is different we do the proof for each canonical model separately .

E.2.1 proof for the case $\chi = +$

Lemma E.2.1. *The MSE map $\Psi(\sigma^2)$ is a concave function of σ^2 for the case $\chi = +$.*

Proof. First, recall the expression for the MSE map

$$\Psi(\sigma^2) = \mathbb{E}\left\{\left(\eta\left(X + \frac{\sigma}{\sqrt{\delta}}Z; \lambda\sigma, \chi\right) - X\right)^2\right\}. \quad (\text{E.7})$$

We denote by $\partial_1\eta$ and $\partial_2\eta$ the partial derivatives of η with respect to its first and second arguments. Using Stein's lemma and the fact that $\partial_1^2\eta(x; y, \chi) = 0$ almost everywhere, we get

$$\begin{aligned} \frac{d\Psi}{d\sigma^2} &= \frac{1}{\delta} \mathbb{E}\left\{\partial_1\eta\left(X + \frac{\sigma}{\sqrt{\delta}}Z; \lambda\sigma\right)^2\right\} \\ &+ \frac{\lambda}{\sigma} \mathbb{E}\left\{\left[\eta\left(X + \frac{\sigma}{\sqrt{\delta}}Z; \lambda\sigma\right) - X\right]\partial_2\eta\left(X + \frac{\sigma}{\sqrt{\delta}}Z; \lambda\sigma\right)\right\}, \end{aligned} \quad (\text{E.8})$$

where we dropped the dependence of $\eta(\cdot)$ on the constraint χ to simplify the formula. In this case we have $X \geq 0$ almost surely and the threshold function is

$$\eta(x; \lambda\sigma) = \begin{cases} (x - \lambda\sigma) & \text{if } x \geq \lambda\sigma, \\ 0 & \text{otherwise.} \end{cases}$$

As a consequence $\partial_1\eta(x; \lambda\sigma) = -\partial_2\eta(x; \lambda\sigma) = \mathbb{I}(x \geq \lambda\sigma)$ (almost everywhere). This yields

$$\begin{aligned} \frac{d\Psi}{d\sigma^2} &= \left(\frac{1}{\delta} + \lambda^2\right) \mathbb{E}\Phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right) \\ &- \frac{\lambda}{\sqrt{\delta}} \mathbb{E}\phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right). \end{aligned}$$

In order to prove the concavity of $\sigma^2 \mapsto \Psi(\sigma^2)$ first notice that a convex combination of concave functions is concave and so it is sufficient to show the concavity in the case $X = x \geq 0$ deterministically. Next notice that, in the case $x = 0$, $\frac{d\Psi}{d\sigma^2}$ is independent of σ^2 . As a consequence, it is sufficient to prove $\frac{d^2\Psi_x}{d(\sigma^2)^2} \leq 0$ where

$$\delta \frac{d\Psi_x}{d\sigma^2} = (1 + \lambda^2\delta) \Phi\left(\frac{\sqrt{\delta}}{\sigma}(x - \lambda\sigma)\right) - \lambda\sqrt{\delta} \phi\left(\frac{\sqrt{\delta}}{\sigma}(x - \lambda\sigma)\right).$$

Using $\Phi'(u) = \phi(u)$ and $\phi'(u) = -u\phi(u)$, we get

$$\delta \frac{d^2\Psi_x}{d(\sigma^2)^2} = -\frac{x}{2\sigma^3} \left\{ 1 + \frac{\lambda\delta}{\sigma} x \right\} \phi\left(\frac{\sqrt{\delta}}{\sigma}(x - \lambda\sigma)\right) < 0 \quad (\text{E.9})$$

for $x > 0$.

□

Now we turn our attention to the proof of Proposition 4.2.2 for the canonical problem $\chi = +$. Since the $\Psi(\sigma^2)$ function is concave the state evolution phase transition can be simplified to

$$\rho_{SE}(\delta; +, \lambda) = \sup\{\rho : \frac{d\Psi}{d\sigma^2} \Big|_{\sigma^2=0} < 1\}. \quad (\text{E.10})$$

As $\sigma \downarrow 0$, we have $\Phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right) \rightarrow 1$ and $\phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right) \rightarrow 0$ if $X > 0$. Therefore,

$$\begin{aligned} \frac{d\Psi}{d\sigma^2} \Big|_0 &= \left(\frac{1}{\delta} + \lambda^2\right) \rho\delta + \left(\frac{1}{\delta} + \lambda^2\right) (1 - \rho\delta) \Phi(-\lambda\sqrt{\delta}) \\ &\quad - \frac{\lambda}{\sqrt{\delta}} (1 - \rho\delta) \phi(-\lambda\sqrt{\delta}). \end{aligned}$$

The threshold $\rho_{SE}(\delta; +, \lambda)$ is obtained by setting $\frac{d\Psi}{d\sigma^2} \Big|_0 = 1$ which is independent of the distribution of non-zero elements of X . Therefore, for the canonical problem $\chi = +$ the phase transition is independent of the distribution of the non-zero elements of X and is given by,

$$\rho_{SE}(\delta; +, \lambda) = \frac{1 - (\frac{1}{\delta})[(1 + z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - [(1 + z^2)\Phi(-z) - z\phi(z)]}, \quad (\text{E.11})$$

where $z = \lambda\sqrt{\delta}$.

Case $\chi = \pm$

Lemma E.2.2. *The MSE map $\Psi(\sigma^2)$ is a concave function of σ^2 for the case $\chi = \pm$.*

Proof. Recall the definition of soft threshold

$$\eta(x; \lambda\sigma) = \begin{cases} (x - \lambda\sigma) & \text{if } x \geq \lambda\sigma, \\ (x + \lambda\sigma) & \text{if } x \leq -\lambda\sigma, \\ 0 & \text{otherwise.} \end{cases}$$

As a consequence $\partial_1\eta(x; \lambda\sigma) = \mathbb{I}(|x| \geq \lambda\sigma)$ and $\partial_2\eta(x; \lambda\sigma) = -\text{sign}(x)\mathbb{I}(|x| \geq \lambda\sigma)$. This yields

$$\begin{aligned} \frac{d\Psi}{d\sigma^2} &= \left(\frac{1}{\delta} + \lambda^2\right) \mathbb{E}\left\{\Phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right) \right. \\ &\quad \left. + \Phi\left(-\frac{\sqrt{\delta}}{\sigma}(X + \lambda\sigma)\right)\right\} \\ &\quad - \frac{\lambda}{\sqrt{\delta}} \mathbb{E}\left\{\phi\left(\frac{\sqrt{\delta}}{\sigma}(X - \lambda\sigma)\right) + \phi\left(\frac{\sqrt{\delta}}{\sigma}(X + \lambda\sigma)\right)\right\}. \end{aligned}$$

Calculating the second derivative at $X = x$ as before it is easy to see that the second derivative is negative and the function is concave. \square

Now we turn our attention to the proof of Proposition 4.2.2 for the canonical problem $\chi = \pm$. As before we consider the behavior of the derivative of $\Psi(\sigma^2)$ at $\sigma^2 = 0$. By letting $\sigma \downarrow 0$ we obtain

$$\begin{aligned} \frac{d\Psi}{d\sigma^2}\Big|_0 &= \left(\frac{1}{\delta} + \lambda^2\right) \rho\delta + \left(\frac{1}{\delta} + \lambda^2\right) (1 - \rho\delta) 2\Phi(-\lambda\sqrt{\delta}) \\ &\quad - \frac{\lambda}{\sqrt{\delta}} (1 - \rho\delta) 2\phi(-\lambda\sqrt{\delta}), \end{aligned}$$

By setting $\frac{d\Psi}{d\sigma^2}\Big|_0 = 1$ we get the $\rho_{SE}(\delta; \pm)$, which is independent of the distribution of the non-zero elements of X . Considering $z = \lambda\sqrt{\delta}$ we get the following formula for

the phase transition.

$$\rho_{SE}(\delta; \pm, \lambda) = \frac{1 - (\frac{2}{\delta})[(1 + z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - 2[(1 + z^2)\Phi(-z) - z\phi(z)]}. \quad (\text{E.12})$$

Case $\chi = \square$

Finally consider the case of X supported on $[-1, +1]$ with $\mathbb{P}\{X \notin \{+1, -1\}\} \leq \epsilon$. In this case we proposed the following nonlinearity,

$$\eta(x) = \begin{cases} +1 & \text{if } x > +1, \\ x & \text{if } -1 \leq x \leq +1, \\ -1 & \text{if } x \leq -1. \end{cases}$$

Notice that the nonlinearity does not depend on any threshold parameter. Since $\partial_1 \eta(x) = \mathbb{I}(x \in [-1, +1])$,

$$\begin{aligned} \frac{d\Psi}{d\sigma^2} &= \frac{1}{\delta} \mathbb{P}\left\{X + \frac{\sigma}{\sqrt{\delta}}Z \in [-1, +1]\right\} \\ &= \frac{1}{\delta} \mathbb{E}\left\{\Phi\left(\frac{\sqrt{\delta}}{\sigma}(1 - X)\right) - \Phi\left(-\frac{\sqrt{\delta}}{\sigma}(1 + X)\right)\right\}. \end{aligned}$$

As $\sigma \downarrow 0$ we get

$$\left. \frac{d\Psi}{d\sigma^2} \right|_0 = \frac{1}{2\delta}(1 + \rho\delta),$$

whence the local stability condition $\left. \frac{d\Psi}{d\sigma^2} \right|_0 < 1$ yields $\rho_{SE}(\delta; \square) = (2 - \delta^{-1})_+$.

Concavity of $\sigma^2 \mapsto \Psi(\sigma^2)$ immediately follows from the fact that $\Phi(\frac{\sqrt{\delta}}{\sigma}(1 - x))$ is non-increasing in σ for $x \leq 1$ and $\Phi(-\frac{\sqrt{\delta}}{\sigma}(1 + x))$ is non-decreasing for $x \geq -1$. Using the combinatorial geometry result of [42] we get

Theorem E.2.1. *For any $\delta \in [0, 1]$,*

$$\rho_{CG}(\delta; \square) = \rho_{SE}(\delta; \square) = \max\{0, 2 - \delta^{-1}\}. \quad (\text{E.13})$$

E.2.2 Proof of Lemma 5.2.4

Lemma 5.2.4 Suppose that the asymptotic behavior (5.25) holds for the message passing algorithm (4.8) and (4.9). Then x_i^t and z_a^t satisfy the following equations

$$\begin{aligned} x_i^{t+1} &= \eta_t \left(\sum_a A_{ia} z_a^t + x_i^t \right) + o_N(1), \\ z_a^t &= y_a - \sum_j A_{aj} x_j^t + \frac{1}{\delta} z_a^{t-1} \langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1}) \rangle + o_N(1), \end{aligned}$$

where the $o_N(1)$ terms vanish as $N, n \rightarrow \infty$.

Proof. To prove the lemma we substitute (5.25) in the general equations (5.26) and write the Taylor expansion of the latter. The update equation for $z_{a \rightarrow i}^t$ yields

$$z_{a \rightarrow i}^t = y_a - \underbrace{\sum_{j \in [N]} A_{aj} x_j^t - \sum_{j \in [N]} A_{aj} \delta x_{j \rightarrow a}^t}_{z_a^t} + \underbrace{A_{ai} x_i^t}_{\delta z_{a \rightarrow i}^t} + O(1/N)$$

For $x_{i \rightarrow a}^{t+1}$ we have

$$x_{i \rightarrow a}^{t+1} = \eta_t \left(\underbrace{\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi} \delta z_{b \rightarrow i}^t}_{x_i^t} - A_{ai} z_a^t \eta'_t \left(\underbrace{\sum_{b \in \partial i} A_{bi} z_b^t + \sum_{b \in \partial i} A_{bi} \delta z_{b \rightarrow i}^t}_{\delta x_{i \rightarrow a}^t} \right) \right) + O(1/N).$$

In underbraces we have identified the various contributions. Substituting the expression indicated for $\delta x_{i \rightarrow a}^t$, $\delta z_{a \rightarrow i}^t$ we obtain the recursion for x_i^t and z_a^t . In particular x_i^t is updated according to

$$\begin{aligned} x_i^{t+1} &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi} \delta z_{b \rightarrow i}^t \right) + o(1) \\ &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + \sum_{b \in [n]} A_{bi}^2 x_i^t \right) + o(1) \\ &= \eta_t \left(\sum_{b \in [n]} A_{bi} z_b^t + x_i^t \right) + o(1). \end{aligned}$$

For z_a^t we get

$$\begin{aligned}
z_a^t &= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \sum_{j \in [N]} A_{aj}^2 z_a^{t-1} \eta'_{t-1} \left(\sum_{b \in [n]} A_{bj} z_b^{t-1} + \sum_{b \in [n]} A_{aj} \delta z_{a \rightarrow j}^{t-1} \right) + o(1) \\
&= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \frac{1}{n} z_a^{t-1} \sum_{j \in [N]} \eta' \left(\sum_{b \in [n]} A_{bi} z_b^{t-1} + x_i^{t-1} \right) + o(1) \\
&= y_a - \sum_{j \in [N]} A_{aj} x_j^t + \frac{1}{\delta} z_a^{t-1} \langle \eta_{t-1} \left(\sum_{b \in [n]} A_{bi} z_b^{t-1} + x_i^{t-1} \right) \rangle.
\end{aligned}$$

□

E.2.3 Proof of Theorem 4.3.1

Suppose that the derivative of the $\Psi(\sigma^2)$ function at zero is equal to ω and below the phase transition the derivative is less than one we have $\omega < 1$. According to concavity $\Psi(\sigma^2) \leq \omega \sigma^2$. Therefore after t application of the Ψ function to σ_0^2 we get $\sigma_t^2 \leq \omega^t \sigma_0^2$ and therefore we have the exponentially fast convergence of the algorithm as claimed in the theorem.

E.2.4 Proof of Theorem 4.4.2

Lemma E.2.3. *For any λ and σ , $F_{\epsilon, \gamma}(\mu)$ is the least favorable distribution for the soft thresholding risk function, i.e., it achieves the $\sup_{F \in \mathcal{F}_{\epsilon, \gamma}} E_{\mu \sim F} r(\mu, \lambda; \sigma)$.*

Proof. Let $G(\mu)$ denote the distribution of non-zero components. We have

$$E(r(\mu, \lambda; \sigma)) = (1 - \epsilon) E(r(0, \lambda; \sigma)) + \epsilon E_{\mu \sim G}(E(r(\mu, \lambda; \sigma) | \mu)).$$

Since the first part of the risk is fixed and does not depend on the distribution G , we ignore it and we just focus on the second term. We call $\Upsilon(\mu; \lambda, \sigma) = E(r(\mu, \lambda; \sigma) | \mu)$.

$$E_{\mu \sim G} \Upsilon(\mu; \lambda, \sigma) = E \Upsilon(\mu^2; \lambda, \sigma) \leq \Upsilon(\gamma^2; \lambda, \sigma). \quad (\text{E.14})$$

The inequality is coming from the Jensen's inequality and concavity of Υ that was

proved in Lemma 4.4.1. The equality is achieved if and only if $P(\mu \in \{\gamma, -\gamma\}) = 1$. We should emphasize that this is just true on the non-zero elements since we conditioned on this event. \square

Corollary E.2.2. *The least favorable distribution is independent of σ and therefore no other thresholding policy can have better phase transition than the minimum risk policy.*

Lemma E.2.4.

$$\lim_{\sigma \rightarrow 0} \frac{\lambda_{MR}^*(\sigma)}{\sigma} = c, \quad (\text{E.15})$$

where c is finite constant.

Proof. By calculating the derivative of the risk function with respect to λ we obtain,

$$\frac{\lambda_{MR}^*}{\sigma} E(1 - \Phi(\frac{\lambda_{MR}^* - \mu}{\sigma}) + 1 - \Phi(\frac{\lambda_{MR}^* + \mu}{\sigma})) = E(\phi(\frac{\lambda_{MR}^* - \mu}{\sigma}) + \phi(\frac{\lambda_{MR}^* + \mu}{\sigma})), \quad (\text{E.16})$$

where ϕ and Φ are the probability density function and distribution function of a normal random variable with mean zero and variance one. If $\frac{\lambda}{\sigma} \rightarrow \infty$ then the left hand side is going to be infinite while the right hand side is 0. Therefore, the limit converges to a finite constant. \square

Theorem E.2.5. *Under the minimum risk thresholding policy λ_{MR}^* the SE phase transition happens at,*

$$\rho_{MR}(\delta) = \max_z \left(\frac{1 - 2/\delta[(1 + z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - 2[(1 + z^2)\Phi(-z) - z\phi(z)]} \right) \quad (\text{E.17})$$

Proof. Consider the following two thresholding policies. The first one is the λ_{MR} and the second one is $\lambda(\sigma) = \beta\sigma$ where β is fixed and does not depend on σ . Also suppose that $\beta = \lim_{\sigma \rightarrow 0} \frac{\lambda_{MR}(\sigma)}{\sigma}$. Call the density evolution mappings for these two thresholdings $\Psi_{MR}(\sigma^2)$ and $\Psi_L(\sigma^2)$. We have,

$$\Psi_{MR}(\sigma^2) \leq \Psi_L(\sigma^2),$$

and

$$\lim_{\sigma \rightarrow 0} \frac{\Psi_{MR}(\sigma^2)}{\Psi_L(\sigma^2)} = 1.$$

Clearly,

$$\rho_{MR}(\delta) > \rho_L(\delta). \quad (\text{E.18})$$

But it has been proved in [38] that $\Psi_L(\sigma^2)$ is a concave function and therefore $\forall \rho > \rho_L$,

$$\lim_{\sigma \rightarrow 0} \frac{\Psi_L(\sigma^2)}{\sigma^2} > 1.$$

Which makes it clear that $\forall \rho > \rho_L$,

$$\lim_{\sigma \rightarrow 0} \frac{\Psi_{MR}(\sigma^2)}{\sigma^2} > 1.$$

In other words this tell us $\rho_{MR}(\delta) > \rho_L(\delta)$. So we basically see that these two methods give us the same phase transitions.

□

Appendix F

Proofs of Chapter 7

F.1 Proof of Theorem 7.2.1

Consider matrix $B \in \mathbb{R}^{2t \times 2t}$ specified by $B_{ii} = 2$, $B_{i(i-1)} = -1$ and $B_{(i-1)i} = -1$. The rest of the elements are equal to zero. This matrix is non-negative definite; therefore we can calculate the square root of this matrix. Define the measurement matrix

$$A = \begin{pmatrix} \sqrt{B} & 0_{2t, N-2t} \\ 0_{n-2t, 2t} & 0_{n-2t, N-2t} \end{pmatrix}, \quad (\text{F.1})$$

Also let y satisfy the following equation $A^*y = [1, 0, \dots, 0]^*$. Put

$$x(\lambda) = \arg \min_x \frac{1}{2} \|y - Ax\|_2^2 + \lambda \|x\|_1.$$

The following fact is clear from the structure of the problem.

Claim 1: $x(\lambda)$ is unique and $\lim_{\lambda \rightarrow 0} x(\lambda) = x^*$ where x^* satisfies $x_i^* = 0$ for $i > 2t$ and $x_i^* = \frac{-i}{2t+1} + 1$ for $1 \leq i \leq 2t$.

According to the above theorem, there exists a λ^* for which $\|x(\lambda) - x^*\|_2^2 \leq \frac{1}{64} \|x^*\|_2^2$. We can now consider the first-order methods for solving $\min_x \frac{1}{2} \|y - Ax\|_2^2 + \lambda^* \|x\|_1$, where (A, y) is the same as before. We will show that there is a lower bound on the distance of the estimates of a first-order method from x^* . Since the algorithm starts at 0, the first derivative is in the direction of A^*e_1 (where e_i is the unit vector

that is zero at every location except at the i^{th} position) and has therefore components in the directions of e_1 and e_2 . Therefore the best thing that a first-order approach can do is to estimate x_1^* and x_2^* correctly and set the rest to zero. With the same reasoning at the ℓ^{th} iteration the best first-order method is able to get $x_1^*, x_2^*, \dots, x_{\ell+1}^*$ right and set the rest to zero. Therefore at iteration t , the mean square error of the estimate from x^* is lower bounded by,

$$\sum_{i=t+1}^{2t} \left(1 - \frac{i}{2t+1}\right)^2 \geq \frac{2t^3}{(2t+1)^2} = \frac{1}{8} \|x^*\|_2^2.$$

Therefore,

$$\|x^t - x(\lambda^*)\|_2 \geq \|x^t - x^*\|_2 - \|x^* - x(\lambda^*)\|_2 \geq .2286 \cdot \|x^*\|_2.$$

Also,

$$\|x(\lambda^*)\|_2 \leq \|x^*\|_2 + \|x^* - x(\lambda)\|_2 \geq \frac{9}{8} \|x^*\|_2.$$

By combining the above two equations we have,

$$\|x^t - x(\lambda^*)\|_2^2 \geq \frac{1}{25} \|x(\lambda^*)\|_2^2.$$

The other claim in the theorem can be proved with the same problem instance and with a very similar method and therefore is not repeated here.

Bibliography

- [1] O. Axelsson. *Iterative solution methods*. Cambridge University Press, first edition, 1996.
- [2] D. Baron, D. Guo, and S. Shamai. A single-letter characterization of optimal noisy compressed sensing. *Proc. of the 47th Annual Allerton Conference*, 2009.
- [3] J. Barzilai and J. M. Borwein. Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, 8:141–148, 1988.
- [4] M. Bayati and A. Montanri. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 2010. submitted.
- [5] M. Bayati, D. Shah, and M. Sharma. Max-product for maximum weight matching: convergence, correctness, and lp duality. *IEEE Transactions on Information Theory*, 54(3):1241–1251, 2008.
- [6] A. Beck and M. Teboulle. A fast iterative shrinkage thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.
- [7] S. Becker, J. Bobin, and E. Candès. NESTA: a fast and accurate first-order method for sparse recovery. 2010. submitted for publication.
- [8] P. Billingsley. *Probability and measure*. John Wiley and Sons, 3 edition, 1995.

- [9] J. Bioucas-Dias and M. Figueiredo. A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Transactions on Image Processing*, 16:2992–3004, 2007.
- [10] J. D. Blanchard, C. Cartis, and J. Tanner. The restricted isometry property and l_q -regularization: Phase transitions for sparse approximation. submitted.
- [11] T. Blumensath and M. E. Davies. Iterative thresholding for sparse approximations. *Journal of Fourier Analysis and Applications, special issue on sparsity*, 14(5):629–654, 2008.
- [12] T. Blumensath and M. E. Davies. How to use the iterative hard thresholding algorithm. *Proc. SPARS*, 2009.
- [13] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.
- [14] J. Bobin, J. L. Starck, J. Fadili, Y. Moudden, and D. L. Donoho. Morphological component analysis: An adaptive thresholding strategy. *IEEE Transactions on Image Processing*, 16(11):2675–2681, 2007.
- [15] J. Bobin, J. L. Starck, and R. Ottensamer. Compressed sensing in astronomy. *IEEE Journal of Selected Topics in Signal Processing*, 2(5):718–726, 2008.
- [16] S. Boyd, N. Parikh, E. Chu, B. Peleato, , and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *to be submitted*, 2010.
- [17] S. Boyd and L. Vanderberghe. *Convex optimization*. Cambridge University Press, 2004.
- [18] K. Bredies and D. Lorenz. Linear convergence of iterative soft-thresholding. *Journal of Fourier Analysis and Applications*, 14:813–837, 2008.
- [19] E. Candès. The restricted isometry property and its implications for compressed sensing. *Compte Rendus de l'Academie des Sciences*, 346:589–592, 2008.

- [20] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, February 2006.
- [21] E. Candès and T. Tao. Rejoinder: the dantzig selector: statistical estimation when p is much larger than n . *Annals of Statistics*, 35:2392–2404, 2007.
- [22] R. Chartrand. Exact reconstructions of sparse signals via nonconvex minimization. *IEEE Signal Processing Letters*, 14:707–710, 2007.
- [23] S. Chen, S. A. Billing, and W. Lue. Orthogonal least squares methods and their application to nonlinear system identification. *International Journal of Control*, 50(5):1873–1896, 1989.
- [24] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1998.
- [25] A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k -term approximation. *IGPM Report*, July 2006.
- [26] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200, 2005.
- [27] W. Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Transactions on Information Theory*, 55(5):2230–2249, 2009.
- [28] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 75:1412–1457, 2004.
- [29] M. A. Davenport and M. B. Wakin. Analysis of orthogonal matching pursuit using the restricted isometry property. *IEEE Transactions on Information Theory*, 2010. to appear.

- [30] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. *Amer. Math. Soc. Lecture: "Math challenges of the 21st century"*, 2000.
- [31] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):489–509, April 2006.
- [32] D. L. Donoho. High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discrete and Computational Geometry*, 35(4):617–652, 2006.
- [33] D. L. Donoho, I. Drori, Y. Tsaig, and J. L. Starck. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. *Stanford Statistics Department Technical Report*, 2006.
- [34] D. L. Donoho and M. Elad. Maximal sparsity representation via minimization. *Proceedings of National Academy of Sciences*, 100:2197–2202, March 2003.
- [35] D. L. Donoho, M. Elad, and V. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52:6–18, January 2006.
- [36] D. L. Donoho and I. M. Johnstone. Minimax risk over l_p balls. *Probability Theory and Related Fields*, 99:277–303, 1994.
- [37] D. L. Donoho, I. M. Johnstone, J. C. Hoch, and A. S. Stern. Maximum entropy and the nearly black object. *Journal of the Royal Statistical Society, Series B (Methodological)*, 54(1):41–81, 1992.
- [38] D. L. Donoho, A. Maleki, and A. Montanari. Message passing algorithms for compressed sensing. *Proceedings of National Academy of Sciences*, 106(45):18914–18919, 2009.
- [39] D. L. Donoho, A. Maleki, and A. Montanari. Construction of message passing algorithms for compressed sensing. *submitted to IEEE Transactions on Information Theory*, 2010.

- [40] D. L. Donoho, A. Maleki, and A. Montanari. Message passing algorithms for compressed sensing: I. motivation and construction. *Proc. of Information Theory Workshop*, 2010.
- [41] D. L. Donoho, A. Maleki, and A. Montanari. Noise sensitivity phase transition. *IEEE Transactions on Information Theory*, 2010. submitted.
- [42] D. L. Donoho and J. Tanner. Counting the faces of randomly projected hypercubes and orthants with applications. *Discrete and Computational Geometry*, to appear.
- [43] D. L. Donoho and J. Tanner. Precise undersampling theorems. *IEEE Transactions on Information Theory*. submitted for publication.
- [44] D. L. Donoho and J. Tanner. Neighborliness of randomly-projected simplices in high dimensions. *Proceedings of the National Academy of Sciences*, 102(27):9452–9457, 2005.
- [45] D. L. Donoho and J. Tanner. Counting faces of randomly projected polytopes when the projection radically lowers dimension. *Journal of American Mathematical Society*, 22:1–53, 2009.
- [46] D. L. Donoho and J. Tanner. Observed universality of phase transitions in high-dimensional geometry, with applications in modern signal processing and data analysis. *Philosophical Transactions of the Royal Society A*, 367(1906):4273–4293, 2009.
- [47] D. L. Donoho and Y. Tsaig. Fast solution of ℓ_1 -norm minimization problems when the solution may be sparse. *IEEE Transactions on Information Theory*, 54(11):4789–4812, November 2008.
- [48] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, 2008.

- [49] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32:407–499, 2004.
- [50] M. Elad, B. Matalon, J. Shtok, and M. Zibulevsky. A wide-angle view at iterated shrinkage algorithms. *Proc. SPIE (Wavelet XII)*, August 2007.
- [51] M. Elad, B. Matalon, and M. Zibulevsky. Image denosing with shrinkage and redundant representations. *Proceedings of the IEEE computer Society Conference on Computer Vision and Pattern Recognition*, 2006.
- [52] M. Elad, J. L. Starck, P. Querre, and D. L. Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Applied and Computational Harmonic Analysis*, 19:340–358, November 2005.
- [53] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of American Statistical Association.*, 96(456):1348–1360, 2001.
- [54] R. Fergus, A. Torralba, and W. T. Freeman. Random lens imaging. *MIT Technical Report MIT-CSAIL-TR-2006-058*, September 2006.
- [55] M. Figueiredo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 16(12):2980–2991, 2007.
- [56] M. Figueiredo and R. Nowak. An em algorithm for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 12(8):906–916, 2003.
- [57] M. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics of Signal Processing*, 1(4):586–598, 2007.
- [58] M. Fornasier and H. Rauhut. Iterative thresholding algorithms. *Applied and Computational Harmonic Analysis*, 25(2):187–208, 2008.

- [59] S. Foucart. Sparse recovery algorithms: Sufficient conditions in terms of restricted isometry constants. 2010. submitted.
- [60] R. G. Gallager. *Low-density parity-check codes*. MIT Press, Cambridge, Massachusetts, 1963. Available online at <http://web./gallager/www/pages/ldpc.pdf>.
- [61] A. Y. Garnaev and E. D. Gluskin. On widths of the euclidean ball. *Soviet Mathematics Doklady*, 30:200–203, 1984.
- [62] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin. One sketch for all: fast algorithms for compressed sensing. *Proc. STOC*, pages 237–246, 2007.
- [63] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing*, 45:600–616, March 1997.
- [64] E. Hale, W. Yin, and Y. Zhang. Fixed point continuation method for ℓ_1 minimization with application to compressed sensing. *Rice University Technial Report TR07-07*, 2007.
- [65] K. K. Herrity, A. C. Gilbert, and J. A. Tropp. Sparse approximation via iterative thresholding. *Proc. ICASSP*, 3:624–627, May 2006.
- [66] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, first edition, 1985.
- [67] I. M. Johnstone. Function estimation and gaussian sequence models. *Mono-graph*, 2010.
- [68] Y. Kabashima, T. Wadayama, and T. Tanaka. A typical reconstruction limit for compressed sensing based on lp-norm minimization. *Journal of Statistical Mechanics*, 2009.
- [69] B. S. Kashin. Diameters of certain finite-dimensional sets in classes of smooth functions. *Izv. Akad. Nauk SSSR*, 41(2):334–351, 1977.

- [70] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. A method for large-scale l_1 -regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):606–617, December 2007.
- [71] M. Ledoux and M. Talagrand. *Probability in Banach spaces*. Springer, second edition, 2002.
- [72] Y. Lu, A. Montanari, B. Prabhakar, S. Dharmapurikar, and A. Kabbani. Counter braids: A novel counter architecture for per-flow measurement. In *SIGMETRICS*, Annapolis, June 2008.
- [73] M. Lustig, D. L. Donoho, and J. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, December 2007.
- [74] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly. Compressed sensing MRI. *IEEE Signal Processing Magazine*, 2008.
- [75] A. Maleki. Coherence analysis of iterative thresholding algorithms. *To be submitted to IEEE Transactions on Information Theory; Partially published in Proc. of 44th Allerton Conference*, 2010.
- [76] A. Maleki and D. L. Donoho. Optimally tuned iterative thresholding algorithm for compressed sensing. *IEEE Journal of Selected Areas in Signal Processing*, April 2010.
- [77] A. Maleki and A. Montanari. Analysis of approximate message passing algorithm. *44th Annual Conference on Information Sciences and Systems*, 2010.
- [78] S. Mallat and S. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, December 1993.
- [79] M. Mézard and A. Montanari. *Information, physics, computation: Probabilistic approaches*. Cambridge University Press, 2008.

- [80] D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2008.
- [81] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *IEEE Journal of Selected Topics in Signal Processing*, 2009.
- [82] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$. *Doklady AN USSR(translated as Soviet math Docl.)*, 269, 1983.
- [83] Y. Nesterov. *Introductory lectures on convex optimization*, volume 87. Kluwer Academic Publishers, 2004.
- [84] Y. Nesterov. Gradient methods for minimizing composite objective function. *CORE Report*, 2007.
- [85] H. Nyquist. Certain topics in telegraph transmission theory. *Trans. AIEE*, 47:617–644, 1928.
- [86] M. R. Osborne, B. Presnell, and B. A. Turlach. A new approach to variable selection in least squares problems. *IMA Journal on Numerical Analysis*, 20:389–404, 2000.
- [87] Y. C. Pati, R. Rezaeiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. *Proc. 27th Asilomar Conference on Signals, Systems and Computers*, A. Singh, ed., IEEE Comput. Soc. Press, 1993.
- [88] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, San Francisco, 1988.
- [89] S. Rankan, A. K. Fletcher, and V. K. Goyal. Asymptotic analysis of map estimation via the replica method and applications to compressed sensing. *submitted to IEEE Transactions on Information Theory*, 2010.

- [90] T. J. Richardson and R. Urbanke. *Modern coding theory*. Cambridge University Press, Cambridge, 2008. Available online at <http://lthcwww.epfl.ch/mct/index.php>.
- [91] S. Ross. *Stochastic processes*. John Wiley and Sons, 2 edition, 1996.
- [92] S. Sardy, A. G. Bruce, and P. Tseng. Block coordinate relaxation methods for nonparametric wavelet denoising. *Journal of Computational and Graphical Statistics*, 9:361–379, 2000.
- [93] S. Sarvotham, D. Baron, and R. Baraniuk. Compressed sensing reconstruction via belief propagation. Preprint, 2006.
- [94] K. Schnass and P. Vandergheynst. Average performance analysis for thresholding. *IEEE Signal Processing Letters*, 14(11):828–831, 2007.
- [95] C. E. Shannon. Communication in the presence of noise. *Proc. Institute of Radio Engineers*, 37(1):10–21, 1949.
- [96] J. L. Starck, M. Elad, and D. L. Donoho. Redundant multiscale transforms and their application for morphological component analysis. *Journal of Advances in Imaging and Electron Physics*, 132:287–348, 2004.
- [97] J. L. Starck, M. Elad, and D. L. Donoho. Image decomposition via the combination of sparse representations and a variational approach. *IEEE Transactions on Image Processing*, 14(10):1570–1582, October 2005.
- [98] C. Stein. *Approximate computation of expectations*, volume 7. Institute of Mathematical Statistics, 1986.
- [99] T. Strohmer and R. W. Heath. Grassmannian frames with applications to coding and communication. *Applied and Computational Harmonic Analysis*, 14(3):257–275, 2003.
- [100] W. Tang, J. Ma, and F. J. Herrmann. Optimized compressed sensing for curvelet-based seismic data reconstruction. *Preprint*, 2009.

- [101] T. Tao. Lecture notes on random matrix theory. 2007.
- [102] D. J. Thouless, P. W. Anderson, and R. G. Palmer. Solution of ‘solvable model of a spin glass’. *Phil. Mag.*, 35:593–601, 1977.
- [103] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series A*, 58(1):267–288, 1996.
- [104] J. A. Tropp. Just relax: Convex programming methods for identifying sparse signals. *IEEE Transactions on Information Theory*, 51(3):1030–1051, 2006.
- [105] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.
- [106] E. van den Berg and M. P. Friedlander. Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing*, 31(2):890–912, 2008.
- [107] L. Vanderberghe. Optimization methods for large-scale systems. *Lecture notes of the course EE236C at University of California Los Angeles*, 2009.
- [108] R. Vershynin. Nonasymptotic theory of random matrices. *Lecture notes for Math 280*.
- [109] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Map estimation via agreement on trees: message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005.
- [110] E. T. Whittaker. On the functions which are represented by the expansions of the interpolation theory. *Proc. Royal Soc. Edinburgh, Sec. A*, 35:181–194, 1915.
- [111] E. Wigner. Characteristic vectors of bordered matrices with infinite dimensions. *Annals of Mathematics*, 62:548–564, 1955.
- [112] E. Wigner. On the distribution of the roots of certain symmetric matrices. *Annals of Mathematics*, 67:325–328, 1958.

- [113] S. Wright and M. Figueiredo R. Nowak. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.
- [114] W. Xu and B. Hassibi. On sharp performance bounds for robust sparse signal recovery. *Proc. of the IEEE International Symposium on Information Theory*, pages 493497,, 2009.
- [115] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for ℓ_1 -minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.
- [116] F. Zhang and H. Pfister. On the iterative decoding of high-rate ldpc codes with applications in compressed sensing. *arXiv:0903.2232v2*, 2009.