Data Driven Algorithms for the Estimation of Low Rank Signals in Structured Subspaces

by

Raj Tejas Suryaprakash

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering:Systems)
in The University of Michigan
2017

Doctoral Committee:

Associate Professor Raj Rao Nadakuditi, Chair
Professor Jinho Baik
Professor Alfred O. Hero III
Dr. Muralidhar Rangaswamy, Air Force Research Labs

Raj Tejas Suryaprakash

rajtejas@umich.edu

ORCID iD: 0000-0002-3567-019X

© Raj Tejas Suryaprakash 2017

For my parents, Usha and Nagarajarao Suryaprakash

ACKNOWLEDGEMENTS

I am deeply grateful to my advisor, Prof. Raj Nadakuditi. I cannot fully express how thankful I am for his support and guidance. He allowed me the freedom to pursue my ideas, and patiently encouraged me whenever I got stuck. His knowledge, insight, and enthusiasm for research are inspiring. Looking back, I realize that in addition to academics, I have received an invaluable education in how to look for the conceptual problem at the heart of a complex problem, become a better researcher, present my work well, and so much more.

I thank my committee members, Prof. Alfred Hero, Prof. Jinho Baik, and Dr. Muralidhar Rangaswamy, for their valuable feedback. I also thank Dr. Rangaswamy for hosting me at AFRL, and for the illuminating conversations about my research: they helped to better understand the case of multirank signals and improve upon my results.

I thank the faculty of EE:Systems, and my teachers in other departments at Michigan, for the several engaging courses I took over the years. I am fortunate to have attended such a good school and learned from them.

I thank Dr. Milutin Pajovic for hosting me at MERL. I enjoyed our discussions throughout my internship and came away with a better understanding of several concepts. I also thank Dr. Phil Orlik and Dr. KJ Kim for their thought-provoking inputs during our meetings.

I thank Dr. Sandeep Gogineni, Dr. Pawan Setlur, and Dr. Tariq Qureshi at AFRL for spending their valuable time with me, and giving me many helpful inputs about my research.

I thank my fellow members in the Nadakuditi group: Curtis Jin, Nick Asendorf, Brian Moore, Himanshu Nayar and Arvind Prasadan. Curtis, thank you for teaching me how to run simulations, helping me with Quals preparation, and for the many other times that you helped me. Nick, thank you for interesting discussions about missing data. Brian, thank you for your help as I was beginning my foray into Kronecker products.

I thank David Hong, Arun Padakandla, Deepanshu Vasal, Farhad Shirani, and Mohsein

Heidari for the many interesting conversations. David, I will fondly remember our impromptu whiteboard discussions that would be sparked by questions in passing.

I thank Maruthi Ravichandran, Srinagesh Sharma, Megha Acharya, Krishna Chaitanya Garikipathi, Manu Akula, Purvaj Rajan, Deepika Mutukuri, and Kirithika Rajendran for making my stay at Ann Arbor pleasant and fun. Maruthi and Nagesh, thank you for being patient roommates, and the lengthy chats about everything under the sun.

I thank my friends from college: Chaitanya Hegde, Merin Mandanna, Akhila Athresh, Kruthi Sabnis, Vinay Hegde, and Siri Srinivas. Chaiti, thank you for being an enthusiastic fellow traveler. I thank my friends from school: Abhishek Ramachandra, Narasimha Datta and Karthik Shivaram. Karthik, thank you for joining me on a memorable drive across the US. Abhi, thank you for humoring my monologues.

I thank the staff in EECS and the Systems Office: Becki Turanksi, Shelly Feldkamp, Ann Pace, Judi Jones, Anne Rhoades, Jose Antonio-Rubio and Kristen Thornton, for their immense help with the administrative processes.

Most importantly, I thank my parents, Usha and Nagarajarao Suryaprakash, for their love, support, and constant encouragement, without which none of this would have been possible. I am grateful to them for exposing me to the joys of learning. I dedicate this thesis to them.

TABLE OF CONTENTS

DEDICATION ii

ACKNOWLEDGEMENTS iii

LIST OF TABLES vii

LIST OF FIGURES ix

LIST OF APPENDICES xvii

ABSTRACT xviii

CHAPTER

I Introduction 1

II Consistency and MSE Performance of MUSIC Algorithm for Single Source in White Noise with Missing Data 8

II.1 Introduction . . . 8
II.2 System Model and MUSIC Algorithm . . . 11
II.3 Consistency of DOA Estimate . . . 13
II.3.1 Consistency of MUSIC DOA for ULA . . . 13
II.3.2 Consistency for Other Low Coherence Arrays . . . 14
II.3.3 Discussion . . . 14
II.3.4 Proof of Proposition II.1 . . . 15
II.3.5 Proof of Theorem II.1 . . . 17
II.4 Expression for MSE of MUSIC DOA Estimate for ULA . . . 18
II.4.1 Previous Results . . . 23
II.5 Numerical Simulations . . . 24
II.6 Conclusions . . . 29

III Algorithms for Estimation of Rank One Signal in Noise with Double Kronecker Product Structured Singular Vectors **30**
III.1 Introduction . . . 30
III.2 Model and Estimation Problem . . . 32
III.2.1 Connections to Low Rank Tensor Estimation . . . 33
III.3 Van Loan-Pitsianis Rearrangement Operator . . . 33
III.4 New Algorithms for the Estimation of u . . . 35
III.4.1 KPCA2,1 . . . 36
III.4.2 KPCA2,2ℓ . . . 37
III.4.3 KPCA2,2r . . . 38
III.4.4 KPCA2,3 . . . 38
III.5 Algorithms Stability via Matrix Perturbation Theory . . . 39
III.5.1 Stability of KPCA2,1 . . . 41
III.5.2 Stability of KPCA2,2ℓ . . . 43
III.5.3 Stability of KPCA2,2r . . . 44
III.5.4 Stability of KPCA2,3 . . . 45
III.6 Performance Analysis and Limits . . . 45
III.7 Numerical Simulations . . . 48
III.7.1 Application: STAP . . . 48
III.8 Conclusions . . . 56

IV Algorithms for Estimation of Rank One Signal in Noise with Triple Kronecker Product Structured Singular Vectors **57**
IV.1 Introduction . . . 57
IV.2 Model and Estimation Problem . . . 58
IV.2.1 Connections to Low Rank Tensor Decomposition . . . 59
IV.3 New Algorithms for the Estimation of u . . . 59
IV.3.1 Detailed Derivation of KPCA3,4 . . . 64
IV.4 Performance Analysis and Limits . . . 66
IV.5 Numerical Simulations . . . 72
IV.5.1 Application: MIMO STAP . . . 72
IV.6 Conclusions . . . 78

V BeamShrink: An Algorithm for Improved Weighting of SVD Based Beamformers **79**
V.1 Introduction . . . 79
V.2 Data Model and Beamformer . . . 82

V.3 Optimum Weights for Inverse Covariance Estimation 83
V.4 Data Driven Algorithm to Estimate BeamShrink Weights 90
V.5 Numerical Simulations 92
V.6 Conclusions 103

VI TensorShrink: An Algorithm for Improved Denoising of Low Rank Tensor Estimates 105
VI.1 Introduction 105
VI.2 Data Model and Low Rank Estimation 107
VI.3 The ALS Algorithm and Its Initialization 108
VI.4 New Weights for Low Rank Tensor Estimation 113
VI.4.1 TensorShrink for Rank One Signals 113
VI.4.2 TensorShrink for Higher Rank Signals 114
VI.5 Data Driven Estimation of $|\widehat{u}_i^H u_i|^2$ and TensorShrink Algorithm 118
VI.6 TensorBeamshrink: Beamforming via TensorShrink Based Weights 122
VI.7 Numerical Simulations 125
VI.8 Conclusions 131

VII Afterword and Future Work 132

Appendices 134

References 170

LIST OF TABLES

TABLE

4.1 Definition of quantities in Table 4.3 69
4.2 Inner-product-squared estimation accuracy of KPCA3 algorithms. 70
4.3 Critical SNR for informative estimation in KPCA3 algorithms. 71

LIST OF FIGURES

FIGURE

1.1 Signal impinging on a uniform linear array whose elements are d apart from direction θ 1

1.2 Empirical MSE of the MUSIC DOA estimates, for different values of SNR and p. We see that there is a sharp phase transition between the region where MUSIC DOA produces consistent estimates of θ, and there the estimate is equivalent to random guessing. The white curve $\sigma_{\mathsf{crit}}(p)$ which arises out of our analysis accurately predicts the phase transition boundary. 3

1.3 Beampattern vs sweep of angle at receiver for a MIMO STAP system. The system $\widehat{u}$, the estimate of $u = a \otimes b \otimes c$ to compute a beamformer which suppresses power at the interferer, while maintaining power level at the target. We see that $\widehat{u}$ computed using KPCA3,1, which accounts for the Kronecker product structure , results in a much deeper notch depth at the clutter location compared to the SVD based estimate. 5

1.4 Subplot (a) shows the oracle beampattern for a system with 20 clutter sources (white x's) and one target (black o). We can see the performance gain due to the BeamShrink algorithm by comparing the SVD based beampattern in subplot(b), and BeamShrink based beampattern in subplot(c) with subplot(a). 7

2.1 A distributed array, in which communication constraints make it prohibitive to transmit all but a fraction p of he observed data values to the centralized processor which then computes the DOA. 10

2.2 Plot of $2m\,\Re\{\widehat{u}^H q\}^2$ against directions of arrival θ, for a system with 300 sensors spaced 1/3 wavelengths apart, and 200 snapshots, where $u^\perp$ is given by (2.21a). The 'empirical' plots correspond to generating data via (2.4), estimating $\widehat{u}$ via the SVD of $\widehat{Y}$, and computing the average of $2m\,\Re\{\langle \hat{u}, q\rangle\}^2$ over 250000 trials. The conjectured values in each case are $1-\alpha^2$, where α^2 is computed for the different values of $\sigma = \{2.5, 5\}$ and $p = \{0.7, 1\}$ using (A.1a). We see that for the uniform linear array, irrespective of the direction of arrival, $2m\,\Re\{\widehat{u}^H q\}^2$ is accurately predicted by the conjectured value. . . . 22

2.3 Empirical MSE of DOA estimate versus SNR = σ for $m = 150$ and $n = 60$. Here we set $p = 1$, and pick the number of samples (or snapshots) $m = n/1.5$ so that $c = m/n = 1.5$ in Theorem A.1, so that we are operating in the sample deficient regime. Figure 2.3a corresponds to the case where the sensors are $d = 1/2$ wavelength apart, and Figure 2.3b corresponds to the case where the sensors are $d = 1/3$ wavelengths apart in (2.2). The theoretical prediction for the MSE was plotted by using the analytic expression in (2.16). The empirical values correspond to generating the full data matrix and running MUSIC on it; the regions where there is excellent agreement with the theoretical predictions are where the high-dimensional random matrix theoretic predictions are accurate. The predictions are less accurate as one nears the breakdown point which corresponds to values near SNR $\approx c^{1/4}/\sqrt{p}$. 'LLV predictions' corresponds to the MSE prediction (2.33), as derived in [1]. 25

2.4 Empirical MSE of DOA estimate versus fraction of data observed = p, for two different SNRs $\sigma = 2.5$ and 5. System was moderately sized at $m = 150$ and $n = 100$ so that $c = m/n = 1.5$ in Theorem A.1. The two vertical lines correspond to theoretical breakdown points $p_{\text{crit}} = \sqrt{c}/\sigma^2$. We see that $\sigma = 5$, prediction of MSE is accurate throughout, but at low SNR, $\sigma = 2.5$, close to the phase transition region, prediction becomes inaccurate. This is due to the finite dimensions of the simulated system, while the theoretical prediction was derived for the asymptotic regime. 26

2.5 Empirical MSE of DOA estimation for a system with 600 sensors and 400 snapshots, plotted as SNR varies from $0.1-5$ and fraction of observed entries varies from $0.1-1$. The white line is a plot of the phase transition boundary σ_{crit} as a function of p. The phase transition boundary prediction is accurate for the entire range of SNR and p. Due to the finite dimensions of the system, the we see the phase transition boundary in this plot has some width, but asymptotically, it will reduce to a line with coincides with $\sigma_{\mathsf{crit}}(p)$. Below the phase transition boundary, the theoretical MSE returned by the expression (2.16) is ∞, hence the value plotted is the variance of uniformly selecting a DOA in the range of possible values. 27

2.6 Empirical MSE of DOA estimation for a system with 150 sensors and 100 snapshots, plotted as the true DOA varies between 10^o-90^o. The theoretical prediction of the MSE given by (2.16) is plotted in red, and the previous prediction in literature [1] given by (2.33) is plotted as a dashed line. The ratio of observed entries is 0.8. We see that our theoretical predictions accurately predict the MSE for the entire range of DOAs, and throughout, more accurate prediction is achieved (ranging from $0.5-1$dB at 20^o to around 2dB at 90^o) compared to the previous result in the literature. The plot gives us insight into the role of source location and array geometry in DOA estimation performance. 28

3.1 Plot of inner-product-squared accuracy, $|\langle \widehat{u}, u\rangle|^2$ vs SNR (σ), where $\widehat{u} = \widehat{u}_{\mathsf{svd}}$, $\widehat{u}_{\mathsf{kpca2},2\ell}$, $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$, $\widehat{u}_{\mathsf{kpca2},3}$, for $m_1 = 15$, $m_2 = 15$ and $n = 200$ in (3.1). The discrete points are obtained by averaging empirical values over 10^4 trials. The continuous plots are obtained via the theoretical expressions in (3.47). For a given SNR, $\widehat{u}_{\mathsf{kpca2},1}$ has the highest accuracy, greater than $\widehat{u}_{\mathsf{kpca2},2\ell}$ and $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$ which have equal accuracy, greater than $\widehat{u}_{\mathsf{kpca2},3}$. All KPCA2 algorithms have higher accuracy than the SVD estimate. The theoretical predictions are in good agreement with the empirical values. $\sigma_{\text{crit}} = c_{\text{svd}}^{1/4}$ is the phase transition boundary given by (3.41). 51

3.2 Application to STAP: Plot of Notch Depth (dB) given by (3.54) vs SNR (dB), where $\widehat{\Sigma}$ in (3.53) (and thus $\widehat{w}$ in (3.52)) is estimated using $u(\theta_c, f_c)$ (Optimal), and $\widehat{u} = \widehat{u}_{\mathsf{svd}}, \widehat{u}_{\mathsf{kpca2,1}}, \widehat{u}_{\mathsf{kpca2,2}\ell}, \widehat{u}_{\mathsf{kpca2,2r}}, \widehat{u}_{\mathsf{kpca2,3}}\}$. The array look angle is 0^o and the interferer is at 30^o. The dimensions of the system are as in Fig. 3.1. All four versions of the KPCA2 algorithm suppress the interference better than the SVD, e.g. at SNR = 20dB, we obtain 10dB better suppression using KPCA2. KPCA1 has the deepest notch depth, followed by KPCA2,2ℓ, KPCA2,2r and then KPCA2,3. 52

3.3 Application to STAP: Notch depth gain of the KPCA2 algorithms relative to the SVD algorithms vs INR, for the system in Fig. 3.1. We see that all KPCA2 algorithms give us notch depth gain relative to the SVD. KPCA2,1 performs the best, and KPCA2,3 gives the least notch depth gain. As expected from Fig. 3.4, KPCA2,2l and KPCA2,2r give gains which are in between the other two algorithms. All KPCA2 algorithms give 13 dB gain in the high INR regime. 52

3.4 Application to STAP: Notch Depth (dB) of the four KPCA2 algorithms vs interferer location (deg). This plot highlights the relative performance of the KPCA2 algorithms, which may be observed in Fig. 3.1 and 3.2. The dimensions of the system are as in Fig. 3.1. KPCA2,1 has the deepest notch depth at all locations of the interferer, and KPCA2,3 has the shallowest notch depth. The notch depths due to KPCA2,2l and KPCA2,2r are in between those of the other two variants, but either one does not suppress interference uniformly better than the other. For example, KPCA2,2ℓ has a deeper notch depth at -23^o (relative to the array broadside), while KPCA2,2r has a deeper notch depth at -18^o. 53

3.5 Heatmap of the logarithm of empirical inner-product-squared accuracy versus aspect ratio $c_{\mathrm{svd}} = m/n$ and SNR σ, for the system in Fig. 3.1. In the 'hot' region (red), $\widehat{u}_{\mathsf{kpca2,2}\ell}$ is highly correlated with u, and in the cold region (blue), estimation performance of $\widehat{u}_{\mathsf{kpca2,2}\ell}$ is equivalent to choosing a random unit vector on the n dimensional sphere. We see a sharp phase transition boundary between the two regimes, and the theoretical boundary prediction given by (3.47) and (3.41) and accurately predicts the empirically observed boundary. 54

3.6 Application to STAP: Heatmap of the notch depth gain due to KPCA2,1 relative to the SVD, for the same setup as in Fig. 3.1, as a function of interferer location and INR. The target is located at 0^o. We obtain around 15 db gain throughout the region away from the target location. 55

3.7 Plot of empirical estimation performance of KPCA2,1 versus the LowRank-KroneckerPCA (LR-Kron) algorithm introduced in [2], for the same setup as in Fig. 3.1. The plot corresponding to "SVD" is the informativeness of $\widehat{u}_{\mathsf{svd}}$. We see KPCA2,1 and LR-Kron yield identical estimation performance. However, KPCA2,1 prescribes a more computationally direct algorithm involving exactly two SVDs and one rearrangement operation to find the singular vector estimate, whereas LR-Kron is an iterative algorithm which requires two eigendecompositions in each iteration of the algorithm, and the total number of iterations depends upon the rate at which the residual error decreases in an alternate minimization setup. 55

4.1 Plot of estimation performance $|\widehat{u}^H u|^2$ vs INR (dB), for a system with $m_1 = 10, m_2 = 9, m_3 = 8$ and $n = 800$ in (4.1), where $\widehat{u} \in \{\widehat{u}_{\mathsf{svd}}, \widehat{u}_{\mathsf{kpca3,1}}, \ldots, \widehat{u}_{\mathsf{kpca3,8}}\}$. The estimation performance of all KPCA3 algorithms is superior to the canonical SVD estimate. In particular, KPCA3,1 and KPCA3,2 which first compute $\widehat{u}_{\mathsf{svd}}$ and then fit a Kronecker structure to it, give the best performance over KPCA3,3 and KPCA3,8 which exploit the Kronecker structure directly on the observed data. The solid line plots our analytical prediction of estimation performance. We see that our analytical prediction closely matches the empirical performance. 74

4.2 Plot of Notch Depth (dB) at the interferer for a MIMO STAP system, with the same dimensions as in Fig. 4.1. Computing the MVDR filter using KPCA algorithms gives a deeper notch depth compared to forming the filter using the canonical estimate $\widehat{u}_{\mathsf{svd}}$. At high INRs, we get a notch depth gain of around 10 dB using the new algorithms. Among the KPCA algorithms, KPCA3, 1 and KPCA3, 2 give the deepest notch depths, as we expect from the plot in Fig. 4.1. 75

4.3 Beampattern $\mathrm{BP}(\theta_r, \theta_{t,s}, f_{d,s})$ as in (4.19) for the system from Fig. 4.1, plotted against θ_r (in degrees) at true values of $\theta_{t,s} = 20^o$ and $f_{d,s}$ as in (4.17). The target is at 40^o relative to the receiver and interferer is at 10^o. The INR was fixed at 5 dB. The oracle beampattern was computed by using the population manifold vector of the clutter to compute the MVDR beamformer, i.e. $\widehat{u} = u_i$ in (4.18). Similarly, the SVD and KPCA3, 1 plots corresponding to using $\widehat{u} = \widehat{u}_{\mathrm{svd}}$ and $\widehat{u} = \widehat{u}_{\mathrm{kpca3,1}}$ in (4.18). We see that using the KPCA3, 1 estimate gives a deeper notch depth at the interferer location, relative to using the SVD estimate. 76

4.4 Beampattern $\mathrm{BP}(\theta_{r,s}, \theta_t, f_{d,s})$ as in (4.19) for the system from Fig. 4.1, plotted against θ_t (in degrees) at true $\theta_r = 40^o$ and $f_{d,s}$ as in (4.17). In this plot target is at 20^o relative to the transmitter and interferer is at 10^o. This plot is the analogue of Fig. 4.3 at the transmitter. Once again, we see a deeper notch depth in the direction of the interferer when we use $\widehat{u} = \widehat{u}_{\mathrm{kpca3,1}}$ relative to $\widehat{u} = u_{\mathrm{svd}}$ in (4.18). 77

5.1 Plot of average notch depth of SVD based beamformer (5.10), BeamShrink based beamformer (5.28), and beamformer based on exact solution vs SNR σ_1, for a system with $m = 300$, $n = 250$. We see that the BeamShrink based beamformer has a deeper notch depth at lower values of $\sigma_{c,1}$. 94

5.2 Plot of average error $\|R^{-1} - \widehat{R}_z^{-1}\|_F^2$ when $\widehat{R}_z$ uses SVD weights and BeamShrink weights. At low $\sigma_{c,1}$, BeamShrink weights yield smaller error than SVD weights, leading to improved denoising. The average error of the data driven BeamShrink weight is equal to the that achieved by using the exact solution, as suggested by Theorem (V.2). 94

5.3 Plot of SVD and BeamShrink weights as a function of σ_1, for the system in Fig. 5.1. Here we see that the optimum weight for denoising is a shrinkage of the SVD weight. We also see that the data driven weight $\widehat{z}_{\mathsf{bsh}}$ matches the exact solution. 95

5.4 Beampatterns of oracle, SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$, and 3 sources of clutter. 97

5.4 Beampatterns of the SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$ and 3 clutter sources, when the input rank is 7. 99

5.5 Beampattern for SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$, 20 clutter sources of which 12 are informative. 101

5.6 Normalized SINR of SVD based beamformer and BeamShrink beamformers for a system with $m = 300$, $n = 150$, 20 clutter sources, of which 12 are informative. 102

5.7 Difference between SINRs due to BeamShrink and SVD based beamformers for the system in Fig. 5.6. In contrast with Fig. 5.6, the input rank is 25, which is an overestimation of the number of clutter sources. There is a rank mismatch both due to input rank and number of informative components. Since BeamShrink is robust to rank overestimation compared to the SVD, we see that the BeamShrink beamformer yields a higher SINR gain at target: 0.17 dB compared to 0.14 dB in Fig. 5.6c. 103

6.1 Procedure to find estimates $\{\widehat{\sigma}_i, \widehat{s}_i\}$ given $\{\widehat{a}_i, \widehat{b}_i\}$. 109
6.2 Plots of typical normalized RMSE, $\|\mathcal{Y} - \widehat{\mathcal{Y}}\|_F^2 / \|\mathcal{Y}\|_F^2$ versus iterations for four independent runs of the ALS algorithm, for a rank one signal. Here, $m_1 = 32$, $m_2 = 30$, $n = 80$, $\sigma_1 = 2.3$. ALS is terminated when the normalized RMSE dips below a threshold, or changes very little with iterations. When initialized with columnwise KPCA2, ALS converges in fewer iterations than the SVD to the same normalized RMSE. 111
6.3 Plots of typical singular vector estimation accuracy, $|\widehat{u}_1^H u_1|^2$ versus iterations for the setup in Fig. 6.2. The estimation accuracy of $\widehat{u}_1$ initialized through Algorithm 6.3 is higher than when initialized using Algorithm 6.2 during the earlier iterations. 112
6.4 Spectrum of $\widehat{U}_{\mathcal{R},i} = \mathcal{R}(\widehat{u}_{\mathsf{svd,i}})$, *i.e.*, the matrices resulting from the Van Loan-Pitsianis rearrangement of the principal four singular vectors, for a system with a rank 3 signal. Recall that these singular vectors are used to initialize the ALS iterations: the principal left and right singular vectors of $\mathcal{R}(\widehat{u}_{\mathsf{svd,i}})$ give $\widehat{a}_i$ and $\widehat{b}_i$ respectively for the initial iteration. We see that the spectra of the top 3 singular vectors have a signal-plus-noise structure, whereas the spectrum of $\mathcal{R}(\widehat{u}_{\mathsf{svd},4})$ is noise-like. This justifies approximating $|\widehat{a}_i^H a_i|^2$ and $|\widehat{b}_i^H b_i|^2$ using the spectra of $\mathcal{R}(\widehat{u}_{\mathsf{svd,i}})$. 121
6.5 Plot of denoising error vs. input rank, for a system with $m_1 = 20$, $m_2 = 30$ and $n = 400$, over 500 trials. The error of ALS algorithm increases sharply when the input rank is greater than the effective rank, due to the addition of uninformative components to the signal estimate. The use of TensorShrink weights mitigates performance loss due to rank overestimation, and yields much lower denoising error, comparable to weights computed numerically using the oracle signal. 126
6.6 Average weights over 500 trials returned by the ALS and TensorShrink algorithms, compared to the optimum weights computed numerically. The input rank is 30, which is the true rank. We see that the smaller weights are shrunk much more: the lowest weights are close to 0. This gap between the ALS weights and TensorShrink weights leads to the gain in MSE performance at rank = 30 in Fig. 6.5. 127
6.7 TensorShrink weights vs. σ_i for a rank two system. 128
6.8 Beampatterns of beamformers computed using weights returned by the ALS algorithm and TensorBeamshrink. 129

6.9 SINR gain due to TensorBeamshrink relative to the ALS and SVD based beamformers. 130

LIST OF APPENDICES

APPENDIX

A Pertinent Results from Random Matrix Theory **134**
A.1 Limiting Spectrum of $\widetilde{Z}$ . . . 136
A.2 Largest Singular Value and Corresponding Left Singular Vector of $\widetilde{Y}$ . . . 137
A.3 Largest singular value of $\Delta_{\widehat{Y}}$ . . . 141
A.4 Relating the Singular Values and Vectors of $\Delta_{\widehat{Y}}$ and $\widetilde{Y}$ . . . 143
A.5 Justification for Conjecture II.1 . . . 148

B Manifold Vector Scaling for Uniform Linear Arrays **150**

C Proof of Claim III.1 **152**
C.1 Analysis of KPCA2,1 . . . 152
C.2 Analysis of KPCA2,2ℓ . . . 154
C.3 Analysis of KPCA2,2r . . . 156
C.4 Analysis of KPCA2,3 . . . 158

D Proof of Claim IV.1 **159**
D.1 Analysis of KPCA3, 1 . . . 159
D.2 Analysis of KPCA3, 3 . . . 163
D.3 Analysis of KPCA3, 6 . . . 166

ABSTRACT

The estimation of low rank signals in noise is a ubiquitous task in signal processing, communications, machine learning, data science, econometrics, *etc.* A critical step in the estimation of such low rank signals is to identify the signal subspace from the noise subspace via the singular value decomposition (SVD). In this thesis, we focus on both the design of new SVD based algorithms for problems arising in array processing and the theoretical analysis of their performance. Common array processing tasks include the estimation of direction of arrival (DOA) and clutter suppression for enhanced target detection via beamforming.

We study the estimation performance of a Multiple Signal Classification (MUSIC), a popular algorithm for DOA estimation, for a single source system when the observations are corrupted by noise and randomly missing samples. We show that the MUSIC DOA estimate is consistent even in the presence of randomly missing samples for the single source case. We present a unified analysis of MUSIC based on random matrix theory that is valid in both the sample rich and deficient regimes, and derive an analytical expression the mean squared error (MSE) performance of the estimate. Our analysis uncovers a phase transition phenomenon, in terms of a critical SNR, and critical fraction of entries observed, below which the MUSIC algorithm breaks down.

We consider the problem of clutter suppression in ordinary space time adaptive processing (STAP), and multiple input, multiple output STAP (MIMO STAP). For the case of single clutter, we propose new algorithms for improved estimation of clutter subspace, which exploits the double Kronecker product structure (STAP) or triple Kronecker product structure (MIMO STAP) of the underlying singular vector. These algorithms are based on a rearrangement operator introduced by Van Loan and Pitsianis. We use random matrix theory based analysis to quantify the relative estimation performance of the algorithms, and their breakdown points. We discover via simulations and analysis that algorithms which first compute the SVD of the data, and then exploit the Kronecker product structure provide better estimation performance than algorithms which first exploit the Kronecker product structure and then use the SVD.

We consider the problem of dominant mode rejection (DMR) beamforming. We derive optimal (rank one case) and approximate (higher rank case) shrinkage weights for clutter subspace estimates computed via the SVD, for improved inverse covariance matrix estimation. We highlight the role of singular vector informativeness in this shrinkage operation. We also use this idea to derive an approximate shrinkage weighting for improved low rank tensor estimation. We propose a data driven algorithms that compute these shrinkage weights for both beamforming and tensor estimation.

Throughout, we validate our analyses with numerical simulations.

CHAPTER I

Introduction

The estimation of low rank signals in noise is a ubiquitous task in signal processing, communications, machine learning, data science, econometrics, and related fields. It is an extremely general framework which encompasses a variety of applications, such as radar, recommendation system design, channel estimation, biomedical signals analysis, *etc.* The particular application that is the focus of this thesis is signals impinging on sensor arrays.

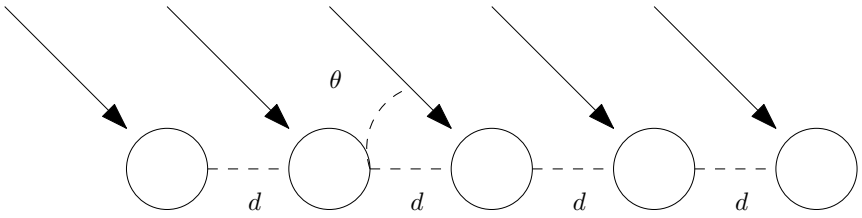


Figure 1.1: Signal impinging on a uniform linear array whose elements are d apart from direction θ

In general, data observed from a low rank signal in noise has the general form

$$Y = \underbrace{UV^H}_{:=S} + Z, \tag{1.1}$$

where the signal matrix UV^H is low rank. However, the physics of the system, constraints, or the nature of an individual problem can impose structure on U or V.

As depicted in Fig. 1.1, let signal from a single source impinge on a uniform linear array of m sensors from an angle θ. The physics of this system imposes the following structure on

U:

$$U = u(\theta) := \frac{1}{\sqrt{m}} \begin{bmatrix} 1 \\ e^{j\,2\,\pi\,d\,\cos(\theta)} \\ \vdots \\ e^{j\,2\,\pi\,d\,(m-1)\,\cos(\theta)} \end{bmatrix}.$$

Here θ is called the Direction of Arrival (DOA), and its estimation is an important task in many signal processing applications. Popular algorithms for DOA estimation include Multiple Signal Classification (MUSIC) and MUSIC-based algorithms [3, 4, 5], root-MUSIC [6, 7], ESPRIT [8, 9], *etc.*

The performance of the MUSIC algorithm for a single source system is the focus of Chapter II. The MUSIC algorithm takes as its input Y, and casts DOA estimation as an optimization problem, whose objective function involves $\widehat{P}_s$, the empirical estimate of the signal subspace projection matrix. The matrix $\widehat{P}_s$ is estimated via $\widehat{u}_{\mathsf{svd}}$, the principal left singular vector of Y as:

$$\widehat{P}_s = \widehat{u}_{\mathsf{svd}}\,\widehat{u}_{\mathsf{svd}}^H.$$

Motivated by distributed array processing scenarios [10], and distributed systems designed to avoid interference which impose bandwidth constraints [11, 12], we assume that a random fraction $p \in (0, 1]$ of the entries in Y are missing, *i.e.*, are 0. Due to the presence of noise, and missing data, there is a performance loss in DOA estimation. Previous approaches to quantifying the performance loss [1] had invoked matrix perturbation theory, which requires that the system be in the sample rich regime. In practice, this is rarely the case.

Our new contribution is a unified analysis of the MUSIC algorithm for the case of a rank one signal in both the sample rich and deficient regimes, and when a random fraction $p \in (0, 1]$ of the entries are missing. Our analysis relies on recent advances in random matrix theory [13, 14, 15]. Firstly, we show that for uniform linear arrays, as well as other array geometries that satisfy a certain "coherence" condition, the MUSIC DOA estimate is consistent. Secondly, we derive an analytical expression for the mean squared error (MSE) performance of the MUSIC algorithm which accounts for the array geometry, as well as the estimation accuracy of the sample singular vector relative to the true, underlying singular vector. Finally, our analysis uncovers a phase transition phenomenon, under which the MUSIC DOA estimate is consistent only if the signal-to-noise ratio (SNR) is above a critical SNR, and p is greater than a critical fraction of observed entries; the critical SNR and critical

sample fraction are simple functions of the system dimensions and sample size. The phase transition phenomenon is shown in Fig. 1.2.

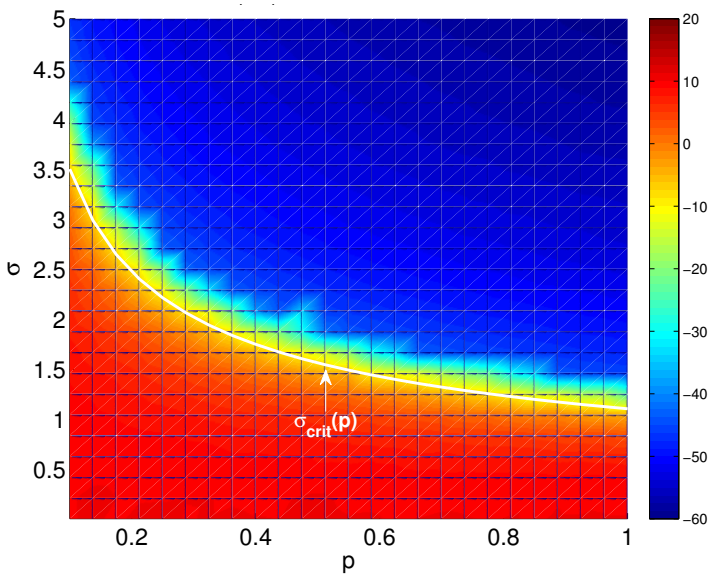


Figure 1.2: Empirical MSE of the MUSIC DOA estimates, for different values of SNR and p. We see that there is a sharp phase transition between the region where MUSIC DOA produces consistent estimates of θ, and there the estimate is equivalent to random guessing. The white curve $\sigma_{\text{crit}}(p)$ which arises out of our analysis accurately predicts the phase transition boundary.

Another important array processing task is space time adaptive processing (STAP), which detects a target in a cluttered radar environment by suppressing clutter. If STAP is carried out in a multiple input multiple output (MIMO) system, it is referred to as MIMO STAP. In the generic low rank system from (1.1), let $U = [u_1 \ \ldots \ u_m] \in \mathbb{C}^{m \times m}$. The STAP system imposes a double Kronecker product structure on the columns of U [16]:

$$u_i = a_i \otimes b_i, \quad a_i \in \mathbb{C}^{m_1}, \ b_i \in \mathbb{C}^{m_2}, \ m = m_1 m_2, \tag{1.2}$$

and the MIMO STAP system imposes a triple Kronecker product structure on the columns of U [17]:

$$u_i = a_i \otimes b_i \otimes c_i \quad a_i \in \mathbb{C}^{m_1}, \ b_i \in \mathbb{C}^{m_2}, \ c_i \in \mathbb{C}^{m_3}, \ m = m_1 m_2 m_3, \tag{1.3}$$

where $\otimes$ denotes Kronecker product [18]. The estimation of rank one signals which have a double and triple Kronecker product structured singular vector is the focus of Chapters III and IV.

In both STAP and MIMO STAP, the canonical estimate of U is via the singular value decomposition (SVD) of the data. Specifically, the columns of U are estimated by computing

the principal left singular vectors of the data matrix. However, the SVD-based estimate is oblivious to the (double or triple) Kronecker product structure of the columns of U. We expect that an estimate that takes into account the Kronecker product of signal singular vectors will yield a more accurate estimate of the underlying singular vector.

In Chapter III, we propose four new algorithms to estimate $U := a \otimes b$, where a and b are some unknown unit-norm vectors, which account for the double Kronecker product structure of the signal's principal singular vector. In Chapter IV, we propose eight new algorithms to estimate $U := u = a \otimes b \otimes c$, which take into account the triple Kronecker product structure of the signal's principal singular vector. Each algorithm involves three SVD computations. All of these algorithms are based on a rearrangement operation introduced by Van Loan and Pitsianis [19].

We also provide an analysis of the relative estimation performance of all the aforementioned algorithms that builds on insights from random matrix theory. Our analysis brings into sharp focus a phase transition boundary, which separates regimes where the algorithms return accurate estimates of the underlying singular vectors and where the algorithms fail. Our simulations and performance analyses reveal that the best performance is achieved by the algorithm which first computes the SVD estimates of the singular vectors, and then utilizes the Kronecker structure to effectively denoise the estimate. Rearranging the data first to exploit the Kronecker structure results in a performance loss.

Crucially, all of the algorithms provide more accurate estimates of the underlying singular vector compared to the principal left singular vector, because they account for the Kronecker product structure. Fig. 1.3 shows how the algorithm performs better than SVD in clutter suppression.

Central to STAP and MIMO STAP is the concept of a beamformer, which suppresses power at clutter while maintaining a peak at the target. In general, beamforming is a ubiquitous task in array processing applications [20, 21, 22, 23]. Chapter V is devoted to improving the performance of SVD based beamformers. Going back to the generic low rank signal model (1.1), If V is random such that $\mathbb{E}\left[V^H V\right] = \Sigma_s^2$, the matrix $U \in \mathbb{C}^{m \times r}$ has rank r, and noise matrix Z is independent of the signal, and has mean 0 and identity covariance, the population covariance matrix of Y is given by

$$R = I + U\Sigma_s^2 U^H.$$

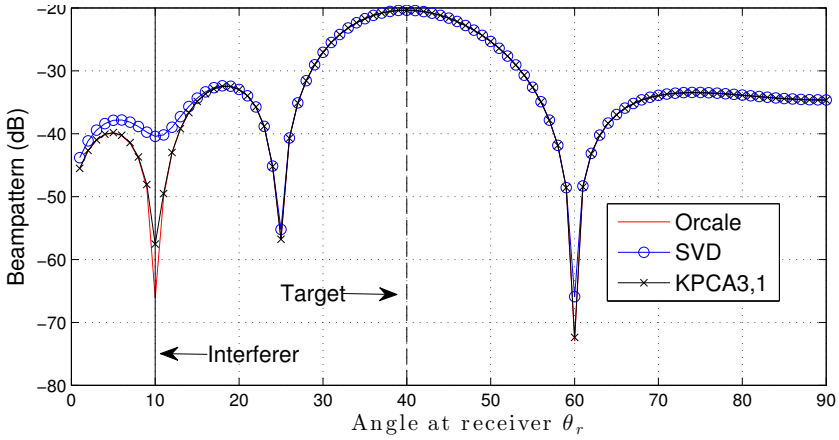


Figure 1.3: Beampattern vs sweep of angle at receiver for a MIMO STAP system. The system $\widehat{u}$, the estimate of $u = a \otimes b \otimes c$ to compute a beamformer which suppresses power at the interferer, while maintaining power level at the target. We see that $\widehat{u}$ computed using KPCA3,1, which accounts for the Kronecker product structure , results in a much deeper notch depth at the clutter location compared to the SVD based estimate.

The canonical estimate of R is given by

$$\widehat{R} = I + \widehat{U}\widehat{\Sigma}_s^2\widehat{U}^H,$$

where $\widehat{U}$ is the matrix containing the top r principal components of Y, and $\widehat{\Sigma}_s$ is a diagonal matrix containing the top r singular values.

The beamformer is a function of the inverse of the covariance matrix. Therefore, the ideal beamformer is a function of R^{-1}, while its estimate is a function of $\widehat{R}^{-1}$. We expect the beamformer performance to improve the closer $\widehat{R}^{-1}$ is to R^{-1}. To that end, we solve the optimization problem

$$z_{\mathsf{bsh}} = \|R^{-1} - (I + \widehat{U}\,\mathrm{diag}(z_1^2, \ldots, z_r^2)\,\widehat{U}^H)^{-1}\|_F^2.$$

In this setup, we have retained the singular vectors computed from the data in the covariance matrix estimate, but we are optimizing for new weights in place of the singular values of the data. Using the weights z_{bsh}, which we term "BeamShrink" weights, makes the estimated inverse covariance matrix closer to the true inverse covariance matrix, relative to the SVD based estimate, in the Frobenius norm sense. Thus we expect the BeamShrink based beamformer to perform better.

Note that we cannot observe all quantities involved in the objective function, since we

cannot observe R. We find the exact solution to the optimization problem, which is also a function of quantities that we cannot observe. However, we show that we can characterize the asymptotic behavior of the BeamShrink weights in the large array, large sample regime, in terms of analytic expressions that arise in random matrix theory. Specifically, these analytic expressions involve transformations of the limiting spectral distribution of the noise only matrix Z. Furthermore, we show that we can consistently estimate the BeamShrink weights in a data driven manner, and propose a data driven algorithm to do so.

The BeamShrink weights are non-convex shrinkage functions on the singular values of the signal. The origin of the shrinkage nature of the BeamShrink weights is the bias in the sample singular value and singular vector estimates from the data. As such, the BeamShrink weights account for the estimation accuracy of the sample singular vectors in a data driven manner. We show how this makes the beamformer computed using BeamShrink weights robust to rank overestimation. Fig. 1.4 showcases the performance gain we get due to BeamShrink, when the signal rank is overestimated.

The Kronecker product structured signals of Chapters III and IV are equivalent to tensor valued signals upon a rearrangement of their elements. The focus of Chapter VI is the estimation of low rank tensor valued signals. Suppose that the signal matrix S in (1.1) gives us$\mathcal{S}$, a rank r tensor of order three upon a rearrangement of its elements. Furthermore, let $\mathcal{S}$ have the following decomposition:

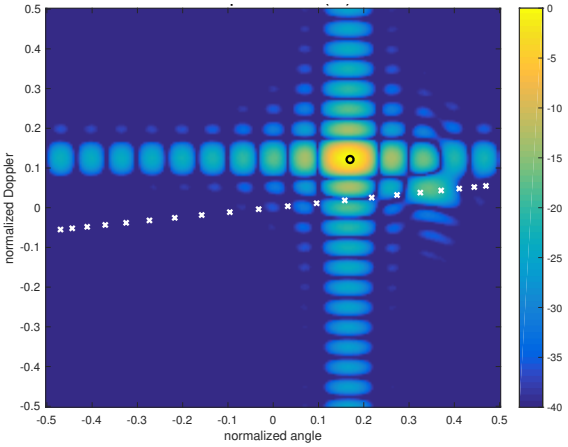
$$\mathcal{S} = \sum_{i=1}^{r} \sigma_i \, a_i \circ b_i \circ c_i,$$

where $\{\sigma_i\} \in \mathbb{R}_+$ are scalars, and $\{a_i\}$, $\{b_i\}$, $\{c_i\}$ are vectors. Let

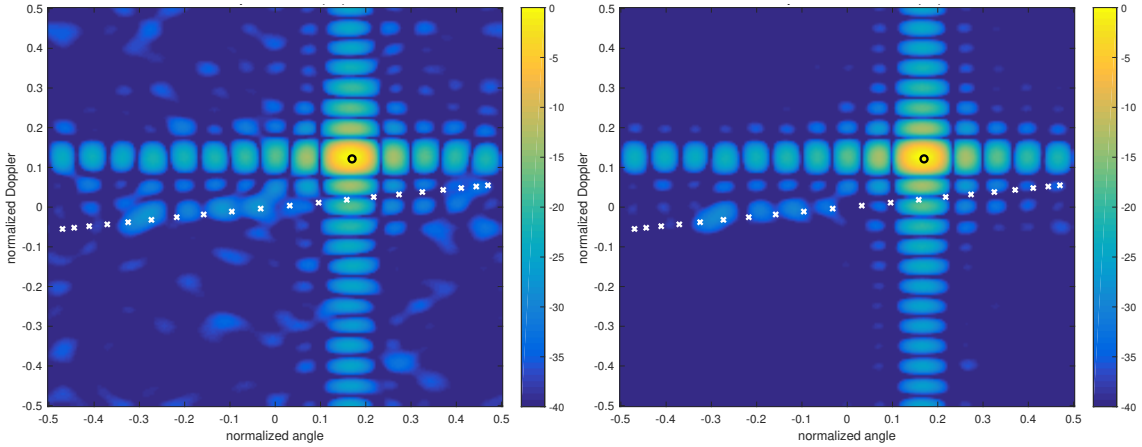
$$\widehat{\mathcal{S}} = \sum_{i=1}^{r} \widehat{\sigma}_i \, \widehat{a}_i \circ \widehat{b}_i \circ \widehat{c}_i,$$

be its estimate, computed from any canonical low rank tensor estimation algorithm, *e.g.*, an alternating least squares based approach. The focus of this chapter is to derive new weights in place of $\{\widehat{\sigma}_i\}$, while retaining $\{a_i\}$, $\{b_i\}$, $\{c_i\}$, such that we obtain an improved estimate of the low rank tensor signal. We term these shrinkage weights as "TensorShrink" weights. The derivation and characterization of optimal TensorShrink weights remains an open problem, but we propose a well justified data driven algorithm that returns suboptimal TensorShrink weights, and show via simulations that this leads to improved low rank tensor estimates.

We also combine the ideas of chapters V and VI, and thus propose an algorithm called



(a) Oracle beampattern



(b) SVD based beamformer

(c) Beamshrink based beamformer

Figure 1.4: Subplot (a) shows the oracle beampattern for a system with 20 clutter sources (white x's) and one target (black o). We can see the performance gain due to the BeamShrink algorithm by comparing the SVD based beampattern in subplot(b), and BeamShrink based beampattern in subplot(c) with subplot(a).

TensorBeamshrink, which leads to the improved performance of beamformers computed using a low rank tensor approximation based approach.

The rest of this thesis is organized as follows. In Chapter II, we analyze the MUSIC algorithm for a rank one signal in the presence of noise and randomly missing data. In Chapter III, we propose and analyze four algorithms for the estimation of a signal with double Kronecker structured singular vectors. In Chapter IV, we present and analyze eight different algorithms for the estimation of a rank one signal with triple Kronecker structured singular vectors. In Chapter V, we present the BeamShrink algorithm. In Chapter VI, we present the TensorShrink and TensorBeamshrink algorithms. Some concluding remarks are presented in Chapter VII.

CHAPTER II

Consistency and MSE Performance of MUSIC Algorithm for Single Source in White Noise with Missing Data

II.1 Introduction

Target localization via direction of arrival (DOA) estimation is an important task in many signal processing applications. Subspace approaches are widely used for this purpose because of their effectiveness [24] as well as their low complexity and robustness relative to maximum-likelihood techniques which are sensitive to signal and noise covariance matrix modeling errors [25]. Popular subspace based algorithms for DOA estimation include Multiple Signal Classification (MUSIC) and MUSIC-based algorithms [3, 4, 5], root-MUSIC [6, 7], ESPRIT [8, 9], and so on.

The MUSIC algorithm, proposed by Schmidt [3], is widely used due to its high resolution capabilities and generality of application [26], and consequently, has been well analyzed in the literature. Properties of MUSIC of interest include DOA estimator consistency and Cramer-Rao bound [27, 28], estimator mean-squared-error (MSE) performance [1, 29], algorithm sensitivity to modeling errors [26, 30, 25, 31, 32], *etc.*

MUSIC takes as its input an $m \times n$ dimensional data matrix, where m is the number of sensors in the sensor array, and the n columns contain snapshots or measurements. It then casts DOA estimation as an optimization problem, whose objective function involves the empirical estimate of the signal subspace projection matrix, which is formed using the rank-k outerproduct of the k leading eigenvectors of the sample covariance matrix. Here k is the number of sources whose signals are impinging on the sensor array. In this work,

we consider the setting single source setting, *i.e.*, $k = 1$. Since we only collect a finite number of snapshots, and the received data is inherently noisy, the sample signal subspace projection matrix is a noisy estimate of the underlying, true projection matrix. This results in a performance loss in estimating the DOA.

Li *et. al.* also considered the problem of quantifying this performance loss in [1]. They derived closed-form expressions for the MSE of subspace based DOA estimators, including MUSIC estimates, as a function of the number of sensors, the number of samples and the signal-to-noise ratio (SNR). Their analytical results employed estimates for signal subspace approximation error based on matrix perturbation theory. Invoking perturbation theory requires that we are operating in a high SNR or large sample regime, where the sample covariance matrix is a consistent estimator of the underlying population covariance matrix. However, in many real world applications, the number of snapshots are of the order of the number of sensors, and therefore, there is a need to characterize the performance of MUSIC in the sample deficient regime [33, 34, 35]. While analysis of the system when the number of sensors and snapshots are finite is essentially intractable [33, 34], we may characterize the DOA estimation performance in the large array regime where the number of sensors and number of snapshots both increase at the same rate, but in which regime the sample covariance matrix is no longer a consistent estimate of the true covariance matrix [36, 37]. Treating the sample covariance matrix as a perturbation of the underlying covariance matrix is not justified in this large array and snapshots regime, since sample covariance matrix may be singular. Here we analyze MUSIC in the presence of noise and missing data in the data matrix, which also contributes to some performance loss.

In the missing data setting, a sample-independent, random fraction $p \in (0, 1]$ of the entries of the snapshot vector are observed. Signal subspace estimation in the presence of such randomly missing entries was also studied in [38, 10]. The missing entry context is motivated by distributed array processing scenarios [10] where it might be prohibitive to collect and transmit all but a (randomly chosen) fraction p of the data entries. For instance, in the distributed setting, system design which avoids channel interference may impose bandwidth constraints [11, 12]. In this setting, each node may transmit a randomly chosen subset of its sensor readings, in a way which avoids complete estimation performance breakdown. Such a situation is depicted in Figure 2.1. The missing data scenario also models random occlusions in data that reach the sensors.

The contributions of this chapter are as follows. First, we show that for a system with a single source, the MUSIC DOA estimate is consistent in the large array regime, even in

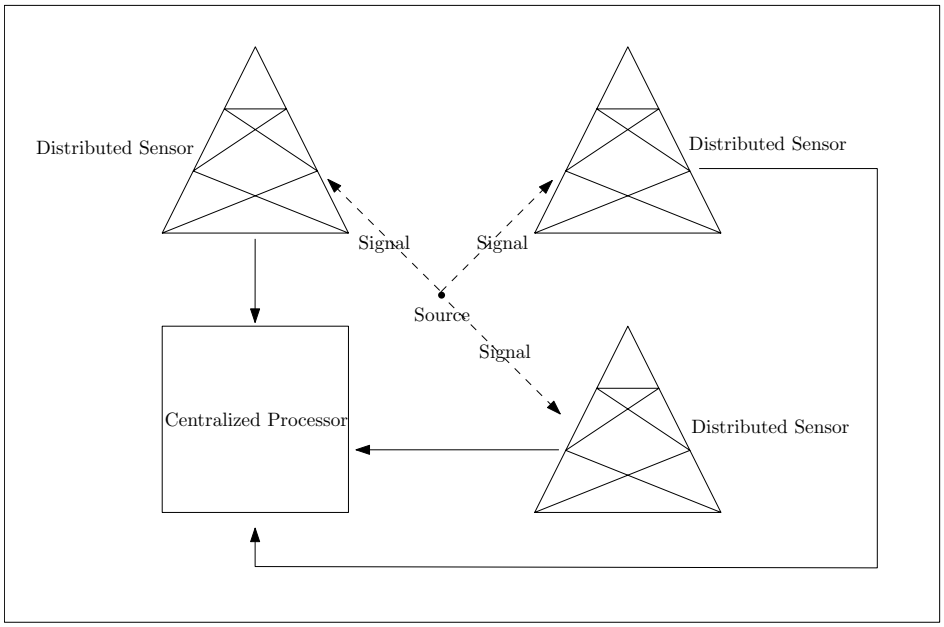


Figure 2.1: A distributed array, in which communication constraints make it prohibitive to transmit all but a fraction p of he observed data values to the centralized processor which then computes the DOA.

the presence of randomly missing data, for a class of arrays which satisfy a low coherence condition. Then, for the uniform linear array, which satisfies the low coherence condition, we provide a unified, analytical expression for the MSE of DOA estimate that is mathematically justified in *both* the sample rich and sample starved regimes. This work also facilitates a precise quantification of the performance loss due to random data compression, and unifies its analysis with those that quantify conventional performance losses. We acknowledge here that there might be other modes of data loss, such as preserving samples but with compression errors, or a reduction in the bits per sample transmitted, but we restrict ourselves to random sample loss.

The expression we uncover for the MSE performance of MUSIC for a single source will also provide insights into the importance of accurate signal subspace estimation, as well as the role of array geometry in the performance of MUSIC. Our analysis is asymptotic in the number of sensors and samples, but extensive numerical simulations show its validity and accuracy for moderately sized systems, and our predictions are 0.5 to 1 dB closer to the empirical simulations than the predictions in [1] for relatively low SNRs (see e.g. Fig. 2.3).

Furthermore, the analysis brings into sharp focus a phase transition phenomenon that reveals the interplay between estimation performance and the system parameters: SNR, the number of sensors and samples and the compression factor. Specifically, we show that

there is a critical fraction of the observed data, p_{crit}, which is a simple function of the largest eigen-SNR, the number of training samples and the number of sensors, below which MUSIC-like algorithms fail to localize the source. Our analysis and results may be used to design systems which operate in a regime where DOA estimates via MUSIC are informative. The identification and precise characterization of this phase transition in the context of the missing data problem is new and goes beyond the analysis in [39].

Our analysis relies on recent results in random matrix theory (RMT) [13, 14, 15, 36, 40, 41] that quantify the bias of the singular vectors of the data or measurement matrix, relative to the true underlying signal (and noise) subspace components. The performance of several subspace based algorithms, including the MUSIC algorithm, relies on exactly the quantity that is characterized by these new results, which facilitates improved accuracy in their analysis. We also extend the results in the RMT literature to unify analysis of the missing data setting. Then, we draw upon the technique used in [1] to obtain an expression for the estimation error which facilitates characterization of the MSE.

We note that the techniques developed here, and the new RMT results presented in this chapter, may be used to analyze the performance of recently developed variants of the MUSIC algorithms (e.g. [5, 34, 35]) or to characterize performance in the presence of model errors as in [42].

The rest of this chapter is organized as follows. The system model and MUSIC algorithm are described in Section II.2. The consistency of the DOA estimate obtained via MUSIC is established in Section II.3. We use these results to derive a closed form expression for the MSE of MUSIC for a single source system in section II.4. Numerical simulations used to validate our predictions are presented in Section II.5 and concluding remarks are presented in II.6.

II.2 System Model and MUSIC Algorithm

Consider a system where a narrowband signal from a single source, located at θ, is incident on a uniform linear array (ULA) having n sensors, which are spaced d wavelengths apart. We collect n independent samples or 'snapshots' of the signal received at each sensor in the array. These snapshot vectors (of length m) may be modeled as [24, 1]

$$y_i = \sigma \underbrace{\frac{a(\theta)}{||a(\theta)||_2}}_{:=u(\theta)} s_i + z_i, \quad i = 1, \ldots, m, \tag{2.1}$$

where σ is the signal-to-noise ratio (SNR), $a(\theta)$ is the array manifold vector or steering vector. The vector $u(\theta)$ has the form

$$u(\theta) = \frac{1}{\sqrt{m}} \begin{bmatrix} 1 \\ e^{j\,2\,\pi\,d\,\cos(\theta)} \\ \vdots \\ e^{j\,2\,\pi\,d\,(m-1)\,\cos(\theta)} \end{bmatrix} \tag{2.2}$$

by virtue of the ULA array geometry. We assume that the elements of z_i have zero mean, variance $1/n$, and bounded higher order moments. Define

$$Y = \begin{bmatrix} y_1 & \cdots & y_n \end{bmatrix},$$

to be the data matrix formed by stacking the snapshots, so that:

$$Y = \underbrace{\sigma\, u(\theta)\, s^H}_{:=S} + Z, \tag{2.3}$$

where s is a unit norm signal vector. We observe $\widehat{Y}$, whose $(i,j)^{\text{th}}$ element is

$$\widehat{Y}_{ij} = \begin{cases} Y_{ij} & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases}. \tag{2.4}$$

Let

$$\widehat{Y} = \widehat{\sigma}\, \widehat{u}\, \widehat{v}^H \tag{2.5}$$

be the SVD of $\widehat{Y}$ truncated to rank one, where $\widehat{\sigma}$ is the principal singular value, and $\widehat{u}$, $\widehat{v}$ are the corresponding left and right singular vectors. The MUSIC algorithm returns $\widehat{\theta}$, an estimate of θ, by solving the optimization problem:

$$\widehat{\theta} = \underset{\phi \in [0,\pi]}{\arg\max} \frac{1}{a(\phi)^H \left[I_n - \widehat{u}\widehat{u}^H \right] a(\phi)}$$

$$\equiv \arg\min_{\phi\in[0,\pi]} a(\phi)^H \left[I_n - \widehat{u}\widehat{u}^H\right] a(\phi),$$

$$\equiv \arg\min_{\phi\in[0,\pi]} ||a(\phi)||_2^2 \left[1 - u(\phi)^H \widehat{u}\widehat{u}^H u(\phi)\right] \tag{2.6a}$$

$$\equiv \arg\max_{\phi\in[0,\pi]}, \underbrace{u(\phi)^H\, \widehat{u}\, \widehat{u}^H\, u(\phi)}_{:=\widehat{\mathcal{P}}(\phi)}. \tag{2.6b}$$

Equation (2.6b) follows from (2.6a) since $||a(\phi)||_2^2 = m \;\forall\; \phi \in [0, \pi]$.

II.3 Consistency of DOA Estimate

Stoica *et. al.* [27, 28] showed that MUSIC DOA estimates were consistent in the fixed m, $n \to \infty$ regime. Mestre *et. al.* [33, 34] analyzed the no-missing-data setting, *i.e.*, when $p = 1$, for a system with $k \geq 1$ sources, and showed that the MUSIC DOA estimates were consistent when $k = 1$ in (2.5) and inconsistent when $p = 1$, $k > 1$ in the $m, n \to \infty$, $m/n \to c \in (0, \infty)$ regime. Here we consider the setting when $p \in (0, 1]$, $k = 1$, and prove that the MUSIC DOA estimates are consistent under some "low coherence" assumptions on the array manifold vector and the signal vector in (2.3).

II.3.1 Consistency of MUSIC DOA for ULA

For simplicity, we begin with the uniform linear array.

Proposition II.1 (Convergence of (2.6b) for ULAs)**.** *Let $u(\theta)$ be manifold vector of the ULA given by (2.2), and assume that s satisfies the* low-coherence condition

$$\|s\|_\infty \leq \eta_v \frac{\log^{C_v}(n)}{\sqrt{n}},$$

for some $\eta_v, C_v > 0$, where $\|s\|_\infty = \max_{i\in\{1,\ldots,n\}} |s_i|$. Let $\widehat{\mathcal{P}}(\phi)$ be defined as in (2.6b) and let

$$\mathcal{P}(\phi) = u(\phi)^H u(\theta) u(\theta)^H u(\phi). \tag{2.7}$$

Then for $p \in (0, 1]$, we have that

$$\sup_{\phi\in[0,\pi]} \left|\widehat{\mathcal{P}}(\phi) - \alpha^2 \mathcal{P}(\phi)\right| \xrightarrow{\text{a.s.}} 0 \;\; \textit{if} \;\; \sigma > \frac{c^{1/4}}{\sqrt{p}},$$

as $m, n \to \infty, m/n \to c \in (0, \infty)$*, where*

$$\alpha^2 = 1 - \frac{c(1 + p\sigma^2)}{p\sigma^2(p\sigma^2 + c)}.$$

Theorem II.1 (DOA estimator consistency)**.** *Assume that the hypotheses of Proposition II.1 hold. Let* $\widehat{\theta}$ *be obtained via* (2.6b)*. Then,*

$$\widehat{\theta} \xrightarrow{\text{a.s.}} \theta \;\; \textit{if} \;\; \sigma > \frac{c^{1/4}}{\sqrt{p}},$$

as $m, n \to \infty$*,* $m/n \to c \in (0, \infty)$*.*

II.3.2 Consistency for Other Low Coherence Arrays

One can generalize the consistency result to non-ULA geometries, provided the vector $u(\theta)$ corresponding to the new geometry satisfies the *low-coherence condition*

$$\|u(\theta)\|_\infty \leq \eta_u \frac{\log^{C_u}(m)}{\sqrt{m}}$$

for some $\eta_u, C_u > 0$, and the limiting beampattern of the array (equivalent to the limit of $\mathcal{P}(\phi)$ in this setting) achieves a unique maximum at the true DOA.

II.3.3 Discussion

We term the assumptions

$$\|u(\theta)\|_\infty \leq \eta_u \frac{\log^{C_u}(m)}{\sqrt{m}}, \quad \|s\|_\infty \leq \eta_v \frac{\log^{C_v}(n)}{\sqrt{n}}$$

as "low-coherence" conditions. A uniform linear array with manifold vector given by (2.2) trivially satisfies the low coherence condition with $\eta_u = 1, C_u = 0$. Many other geometries do as well – see, for e.g. [24]; a complete categorization is beyond the scope of this work.

The low coherence condition for the normalized signal vector s is satisfied with high probability by a vector of i.i.d Gaussians with zero mean and covariance I_m/n with $C_v = 1/2$; see [43]. Signal vectors composed of i.i.d. elements with other distributions (e.g. with

sub-gaussian tails) will also satisfy this condition with high probability; see [44, Theorem 2.5].

The term 'low coherence' as used here arises in the literature of sparse signal processing and matrix completion (e.g. [45, 46, 47, 48]) and is not to be confused with the notion of coherence as used in array signal processing [49]. In our context, a low coherence vector is one that that has small inner product with all canonical, Euclidean basis vectors. The matrix S thus formed from the vectors u and s having low-coherence will have entries that are about the same magnitude. When such a matrix has missing entries, the singular vectors of the matrix formed with zeros in the missing entries location will be correlated with the underlying singular vectors provided that eigen-SNR σ is above a critical threshold σ_{crit} given by

$$\sigma_{\mathsf{crit}} := \frac{c^{\frac{1}{4}}}{\sqrt{p}}. \tag{2.8}$$

In (2.8), c is the (limiting) ratio of the number of array elements and the number of observations. If the eigen-SNR is above this critical threshold then, by Theorem II.1, the MUSIC DOA estimates produced will be asymptotically consistent. When the eigen-SNR is below the critical threshold, then the singular vectors are asymptotically uncorrelated to the true singular vectors and the MUSIC DOA estimates will be inconsistent.

II.3.4 Proof of Proposition II.1

For any $\phi \in [0, \pi]$, express $u(\phi)$ in terms of $u(\theta)$ and q,where q is orthogonal to $u(\theta)$. Specifically, let

$$u(\phi) = \xi_1 u(\theta) + \xi_2 q, \tag{2.9}$$

where $\xi_1 = u(\theta)^H u(\phi)$, and $\xi_2 = q^H u(\phi)$. It is easily seen from (2.6b) that $|\xi_1|^2 = \mathcal{P}(\phi)$. Now,

$$\begin{aligned}
\left\| \widehat{\mathcal{P}}(\phi) - \alpha^2 \mathcal{P}(\phi) \right\|_\infty &= \sup_\phi \left| u(\phi)^H \widehat{u}\widehat{u}^H u(\phi) - \alpha^2 \mathcal{P}(\phi) \right| \\
&= \sup_\phi \left| |u(\theta)^H \widehat{u}|^2 \, |\xi_1|^2 - \alpha^2 |\xi_1|^2 \;\; + |q^H \widehat{u}|^2 \, |\xi_2|^2 + 2\Re\{\xi_1 \xi_2 \, u(\theta)^H \widehat{u} q^H \widehat{u}\} \right| \\
&\leq \sup_\phi \left| |u(\theta)^H \widehat{u}|^2 \, |\xi_1|^2 - \alpha^2 |\xi_1|^2 \right| + \sup_\phi |q^H \widehat{u}|^2 |\xi_2|^2 \\
&\qquad + 2 \sup_\phi \left| |\xi_1| \, |\xi_2| \, |u(\theta)^H \widehat{u}| |q^H \widehat{u}| \right|
\end{aligned} \tag{2.10}$$

where the second line follows from (2.9). We will now show that each term on the right hand side of (2.10) converges to 0 almost surely, in the large array limit. To complete the proof, we will need to characterize the asymptotic behavior of the quantities $|u(\theta)^H \widehat{u}|^2$ and $|q^H \widehat{u}^2|$ which appear in the last line of (2.10). This is where random matrix theory plays an important role: we are able to characterize exactly these required quantities in the large array limit via Theorem A.1. The theorem states that, under certain regularity conditions (termed "low coherence conditions" in Section II.3.3), as $m, n \to \infty$, $m/n \to c \in (0, \infty)$ (see (A.1a), (A.1b))):

$$|u(\theta)^H \widehat{u}|^2 \xrightarrow{\text{a.s.}} \alpha^2,$$
$$|q^H \widehat{u}|^2 \xrightarrow{\text{a.s.}} 0.$$

Since inner products of unit norm vectors are bounded by 1, for all $\phi \in [0, \pi]$,

$$|\xi_1| \leq 1, \;\; |\xi_2| \leq 1, \;\; |u(\theta)^H \widehat{u}| \leq 1.$$

Since $|q^H \widehat{u}|^2 \xrightarrow{\text{a.s.}} 0$, as $m, n \to \infty, m, n \to c$,

$$|q^H \widehat{u}|^2 \, |\xi_2|^2 \xrightarrow{\text{a.s.}} 0,$$
$$\left|\xi_1\right| \left|\xi_2\right| \left|u(\theta)^H \widehat{u}\right| \left|q^H \widehat{u}\right| \xrightarrow{\text{a.s.}} 0.$$

We are now left with the first term in the summation in (2.10):

$$\sup_{\phi} \left| |u(\theta)^H \widehat{u}|^2 \, |\xi_1|^2 - \alpha^2 \, |\xi_1|^2 \right| = \left| \left|u(\theta)^H \widehat{u}\right|^2 - \alpha^2 \right| \sup_{\phi} |\xi_1|^2$$

since $|u(\theta)^H \widehat{u}|$ and α do not depend on ϕ. Since $\sup_{\phi} |\xi_1|^2 \leq 1$, and $|u(\theta)^H \widehat{u}|^2 \xrightarrow{\text{a.s.}} \alpha^2$ from (A.1a), we have that as $m, n \to \infty$, $m/n \to c$,

$$\left| |u(\theta)^H \widehat{u}|^2 \, |\xi_1|^2 - \alpha^2 \, |\xi_1|^2 \right| \xrightarrow{\text{a.s.}} 0.$$

Therefore, the right hand side of (2.10) converges to 0 almost surely in the large array limit. Since $|\xi_1|^2 = \mathcal{P}(\phi)$, we have that as $m, n \to \infty$, $m/n \to c$,

$$\sup_{\phi \in [0, \pi]} \left| \widehat{\mathcal{P}}(\phi) - \alpha^2 \mathcal{P}(\phi) \right| \xrightarrow{\text{a.s.}} 0. \tag{2.11}$$

The above result is analogous to [50, Proposition 4.3.1], which establishes a similar result

when there is no missing data, for a modified version of the empirical MUSIC objective function. We will now show that $\widehat{\theta}$ is a consistent estimator of θ.

II.3.5 Proof of Theorem II.1

Without loss of generality, and for expositional simplicity, prove the result when $d = 1/2$ in (2.2). The proof when $d \neq 1/2$ follows similarly. The proof may be completed using the arguments in [51, Proposition 5]. We follow through the steps of the proof therein for completeness of exposition. Let

$$\eta_m(\phi) = 1 - \mathcal{P}(\phi). \tag{2.12}$$

Then θ is a solution to the equation $\eta_m(\phi) = 0$. Similarly, let

$$\widehat{\eta}(\phi) = 1 - \frac{1}{\alpha^2}\widehat{\mathcal{P}}(\phi), \tag{2.13}$$

where α^2 is given by Proposition II.1. Since $\widehat{\theta}$ minimizes $-\widehat{\mathcal{P}}(\phi)$, we get $|\widehat{\eta}(\widehat{\theta})| \leq |\widehat{\eta}(\theta)|$, and from (2.11), $\widehat{\eta}(\theta) \xrightarrow{\text{a.s.}} \eta_m(\theta)$, we obtain $\widehat{\eta}(\widehat{\theta}) \xrightarrow{\text{a.s.}} 0$. Then,

$$|\eta_m(\widehat{\theta})| \leq |\eta(\widehat{\theta}) - \widehat{\eta}(\widehat{\theta})| + |\widehat{\eta}(\widehat{\theta})| \;\leq \sup_{\phi\in[0,\pi]} |\eta_m(\widehat{\phi}) - \widehat{\eta}(\widehat{\phi})| + |\widehat{\eta}(\widehat{\theta})| \xrightarrow{\text{a.s.}} 0. \tag{2.14}$$

Since $\eta_m(\widehat{\theta})$ depends on $\mathcal{P}(\widehat{\theta})$ via (2.12), we will now evaluate $\mathcal{P}(\widehat{\theta})$. From (2.2) with $d = 1/2$,

$$\mathcal{P}(\widehat{\theta}) = \left| \frac{1}{m} \sum_{i=0}^{m-1} e^{-j\,\pi\, i\,(\Delta\kappa_m)} \right|^2, \tag{2.15}$$

where $\Delta\kappa_m := \cos(\widehat{\theta}) - \cos(\theta)$. The limit of the expression in (2.15) is given by the following lemma.

Lemma II.1. *For a sequence $\Delta\kappa_m$ in a compact subset of $[-2, 2]$, we have*

$$\frac{1}{m}\sum_{i=1}^{m} e^{-j\,\pi\, i\,\Delta\kappa_m} \to \begin{cases} 0 & \text{if } \Delta\kappa_m \nrightarrow 0, \\ 0 & \text{if } \Delta\kappa_m \to 0 \text{ and } m\,|\Delta\kappa_m| \to \infty, \\ e^{j\frac{\beta}{2}}\operatorname{sinc}\left(\frac{\beta}{2}\right) & \text{if } \Delta\kappa_m \to 0 \text{ and } m\,|\Delta\kappa_m| \to \beta. \end{cases}$$

Proof. See [51, Lemma 13], [52].

□

Since $\Delta\kappa_m \in [-2, 2]$, a bounded set, we can extract a convergent subsequence, $\Delta\kappa_{m'}$.

1. If $\Delta\kappa_{m'} \nrightarrow 0$, using Lemma II.1 in (2.15), $\mathcal{P}(\widehat{\theta}) \to 0$. Using this result in (2.12), $\eta_m(\widehat{\theta}) \to 1$, a contradiction with (2.14). Therefore, $\Delta\kappa_m \to 0$.

2. Suppose $m\,|\Delta\kappa_m| \to \infty$. Again, using Lemma II.1 in (2.15), $\mathcal{P}(\widehat{\theta}) \to 0$, leading to a contradiction. Therefore, $m\,|\Delta\kappa_m|$ is also bounded.

3. Then, let $m\,|\Delta\kappa_{m'}| \to \beta$. From Lemma II.1, $\mathcal{P}(\widehat{\theta}) \to \operatorname{sinc}^2\left(\frac{\beta}{2}\right)$. From (2.12), this implies

$$\eta_m(\widehat{\theta}) \to 1 - \operatorname{sinc}^2\left(\frac{\beta}{2}\right), \text{ which is } \begin{cases} > 0 & \text{if} \quad \beta \neq 0, \\ = 0 & \text{if} \quad \beta = 0. \end{cases}$$

   To avoid contradiction with (2.15), $\beta = 0$ so that $m|\Delta\kappa_m| \to 0$. Since every subsequence of $m\,|\Delta\kappa_m| \to 0$, we conclude that $\cos(\widehat{\theta}) \xrightarrow{\text{a.s.}} \cos(\theta)$, and thus $\widehat{\theta} \xrightarrow{\text{a.s.}} \theta$ since $\cos^{-1}(\cdot)$ is a continuous mapping.

Note. Proposition II.1 is critical for the proof of consistency. Once it has been established, one can also complete the proof of Theorem II.1 by a direct application of [53, Theorem 4.1.1].

II.4 Expression for MSE of MUSIC DOA Estimate for ULA

The consistency result in Section II.3 enables us to get a handle on DOA estimation error $\Delta\theta := \widehat{\theta} - \theta$, from which we may obtain an approximate expression for the mean-squared-error (MSE) of MUSIC DOA estimate as $m, n \to \infty$, $m/n \to c \in (0, \infty)$.

Conjecture II.1. *Assume that the hypothesis of Theorem A.1 holds. Then, as* $m, n \to \infty, m/n \to c \in (0, \infty)$,

$$\mathbb{E}\left[\Re\left\{\sqrt{n}\,\widehat{u}^H q\right\}^2\right] \xrightarrow{\text{a.s.}} \frac{1 - \alpha^2}{2},$$

where q *is defined in* (2.9) *and* $\Re\{\cdot\}$ *denotes 'real part of'.*

Claim II.1. *Assume that we observe data according to* (2.4)*, that the hypotheses of Theorem A.1 hold, and that Conjecture II.1 holds. Then, if* $\sigma > c^{1/4}/\sqrt{p}$*, the mean squared error*

of the MUSIC DOA estimate may be computed as

$$\boxed{\mathbb{E}\left[\Delta\theta^2\right] \approx \frac{1-\alpha^2}{2m\delta_1^2\alpha^2}} \tag{2.16}$$

where α^2 *is given by* (A.1a) *and* $\delta_1^2 := \delta_1(\theta)^2 = ||u^{(1)}(\theta)||_2^2 - |\gamma_1(\theta)|^2$, *where* $\gamma_1(\theta) = u(\theta)^H u^{(1)}(\theta)$, *and the superscript* $^{(1)}$ *indicates derivative of the vector.*

Conjecture II.1 has been verified using extensive numerical simulations, as indicated by Figure 2.2, and we provide a justification for it in Section A.5.

We will now derive an expression for DOA estimation MSE when for a system with ULAs (2.2). Recollect from (2.6b) that MUSIC returns as the DOA estimate that angle which maximizes the objective function $\widehat{\mathcal{P}}(\phi)$. Therefore, at $\widehat{\theta}$,

$$\left.\frac{\partial\widehat{\mathcal{P}}(\phi)}{\partial\phi}\right|_{\phi=\widehat{\theta}} := \widehat{\mathcal{P}}^{(1)}(\widehat{\theta}) = 0.$$

Expanding $\widehat{\mathcal{P}}^{(1)}(\widehat{\theta})$ around θ using Taylor Series,

$$\widehat{\mathcal{P}}^{(1)}(\widehat{\theta}) = \widehat{\mathcal{P}}^{(1)}(\theta) + (\Delta\theta)\,\widehat{\mathcal{P}}^{(2)}(\theta) + o\left(\Delta\theta^2\right) = 0. \tag{2.17}$$

This is where the consistency result plays an important part. Since $\widehat{\theta}$ is a consistent estimator of θ, we may neglect the $o(\Delta\theta^2)$ terms to obtain the following approximation for $\Delta\theta$:

$$\Delta\theta = -\frac{\widehat{\mathcal{P}}^{(1)}(\theta)}{\widehat{\mathcal{P}}^{(2)}(\theta)}. \tag{2.18}$$

Note. This technique is was also utilized in [54, 27, 1] for the analysis of MUSIC when the array size is fixed.

Differentiating (2.6b) with respect to ϕ and evaluating the derivatives at θ, we get:

$$\widehat{\mathcal{P}}^{(1)}(\theta) = 2\Re\{u(\theta)^H\widehat{u}\widehat{u}^H u^{(1)}(\theta)\}, \tag{2.19a}$$

$$\widehat{\mathcal{P}}^{(2)}(\theta) = 2\Re\{u(\theta)^H\widehat{u}\widehat{u}^H u^{(2)}(\theta)\} + 2|\widehat{u}^H u^{(1)}(\theta)|^2. \tag{2.19b}$$

Substituting (2.19) into (2.18), we get:

$$\Delta\theta = -\frac{\Re\{u(\theta)^H\widehat{u}\widehat{u}^H u^{(1)}(\theta)\}}{\Re\{u(\theta)^H\widehat{u}\widehat{u}^H u^{(2)}(\theta)\} + |u^{(1)}(\theta)^H\widehat{u}|^2} := \frac{(\Delta\theta)_{\text{num}}}{(\Delta\theta)_{\text{den},1} + (\Delta\theta)_{\text{den},2}}. \tag{2.20}$$

We now decompose the manifold vector derivatives as the sum their components in the signal subspace and orthogonal to it. Let

$$u^{(1)}(\theta) = \gamma_1(\theta)\, u(\theta) + \delta_1(\theta)\, q, \tag{2.21a}$$
$$u^{(2)}(\theta) = \gamma_2(\theta)\, u(\theta) + \delta_2(\theta)\, q. \tag{2.21b}$$

where q is a basis vector of the subspace orthogonal to the signal subspace, and

$$\gamma_1 = u(\theta)^H u^{(1)}, \tag{2.22a}$$
$$\gamma_2 = u(\theta)^H u^{(2)}, \tag{2.22b}$$
$$\delta_1^2 = ||u^{(1)}(\theta)||_2^2 - |\gamma_1|^2, \tag{2.22c}$$
$$\delta_2^2 = ||u^{(2)}(\theta)||_2^2 - |\gamma_2|^2. \tag{2.22d}$$

Equation (2.22a), is obtained when we left multiply (2.21a) by $u(\theta)^H$ to obtain

$$u(\theta)^H u^{(1)}(\theta) = \gamma_1(\theta)\, 1 + \delta_1(\theta)\, 0.$$

Similarly, (2.22c) is obtained by left multiplying (2.21a) by $u^{(1)}(\theta)$. Equations (2.22b) and (2.22d) are obtained analogously using (2.21b). For the rest of this derivation, we suppress the dependence of $\{\gamma_1(\theta), \gamma_2(\theta), \delta_1(\theta), \delta_2(\theta)\}$ on θ, and denote them $\{\gamma_1, \gamma_2, \delta_1, \delta_2\}$ respectively. Furthermore, from (2.22c) and (2.22d), $\delta_1, \delta_2 \in \mathbb{R}$.

We will now relate a few of the above co-ordinates (2.22a)–(2.22d) which will be useful in the forthcoming analysis. From (2.2), $u(\theta)$ is a unit norm vector, so that $u(\theta)^H u(\theta) = 1$. Differentiating both sides of this equation with respect to θ, we obtain

$$2\Re\{u(\theta)^H u^{(1)}(\theta)\} = 2\Re\{\gamma_1\} = 0 \tag{2.23}$$

from (2.22a). Differentiating (2.23) again with respect to θ,

$$2\Re\{u^{(2)}(\theta)^H u(\theta)\} + 2||u^{(1)}(\theta)||_2^2 = 0.$$

Substituting the definitions of γ_2 and δ_1^2 from (2.22b) and (2.22c), we obtain,

$$|\gamma_1|^2 + \Re\{\gamma_2\} = \delta_1^2. \tag{2.24}$$

Now consider $(\Delta\theta)_{\text{num}}$ in (2.20). Substituting the expression from (2.21a) for $u^{(1)}(\theta)$, and

using the notation of (2.22a) and (2.22c),

$$\begin{aligned}(\Delta\theta)_{\text{num}} &= \Re\{\gamma_1\}\,|u(\theta)^H\widehat{u}|^2 + \delta_1\,\Re\{u(\theta)^H\widehat{u}\,\widehat{u}^H q\},\\ &= \delta_1\,\Re\{u(\theta)^H\widehat{u}\,\widehat{u}^H q\},\\ &\approx \delta_1\,\alpha\,\Re\{\widehat{u}^H q\} + o(1),\end{aligned} \tag{2.25}$$

where we have used the fact that $\Re\{\gamma_1\} = 0$ by (2.23), and the approximation

$$u(\theta)^H\widehat{u} \approx \alpha + o(1), \tag{2.26}$$

α given by (A.1a), for the last step. The justification for this approximation is as follows. The objective function of MUSIC, given by $-|u(\phi)^H\widehat{u}|^2$ from (2.6b), depends only on the magnitude of $u(\phi)^H\widehat{u}$ and is invariant to its phase. Therefore, the performance analysis of MUSIC may be carried out assuming $u(\theta)^H u \in \mathbb{R}$. Since from Theorem A.1, $|u(\theta)^H\widehat{u}|^2 \xrightarrow{\text{a.s.}} \alpha^2$, in the regime of our analysis, we can use the approximation in (2.26).

Here, to obtain an MSE expression for DOA, we require rigorous characterization of the second order behaviour of $\widehat{u}^H q$, which appears in (2.25), and in the forthcoming computations. We leave this for future work, and proceed assuming Conjecture II.1 holds.

Squaring (2.25) gives

$$(\Delta\theta)^2_{\text{num}} \approx \delta_1^2\,\alpha^2\,\frac{1-\alpha^2}{2m} + o(1). \tag{2.27}$$

Similarly, we evaluate the denominator of (2.20). Using the expression for $u^{(2)}(\theta)$ from (2.21b) in $(\Delta\theta)_{\text{den},1}$, we get

$$\begin{aligned}(\Delta\theta)_{\text{den},1} &= \Re\{\gamma_2\}^2\,|u(\theta)^H\widehat{u}|^2 + \delta_1\,\Re\{u(\theta)^H\widehat{u}\widehat{u}^H q\}\\ &\approx \Re\{\gamma_2\}\,\alpha^2 + \delta_1\,\alpha\,\Re\{\widehat{u}^H q\} + o(1)\end{aligned} \tag{2.28}$$

where we have used (2.26), and the approximation

$$u(\theta)^H\widehat{u}^2 \approx \alpha^2 + o(1) \tag{2.29}$$

suggested by Theorem A.1 in the regime of our interest. Similar computation for $(\Delta\theta)_{\text{den},2}$ of (2.20), using (2.21a) for $u^{(1)}(\theta)$ and the approximations (2.26) and (2.29) gives

$$(\Delta\theta)_{\text{den},2} = |\gamma_1|^2\,|u(\theta)^H\widehat{u}|^2 + \delta_1^2\,|\widehat{u}^H q|^2 + 2\,\delta_1\,\Re\{\gamma_1\,u(\theta)^H\widehat{u}\,\widehat{u}^H q\}$$

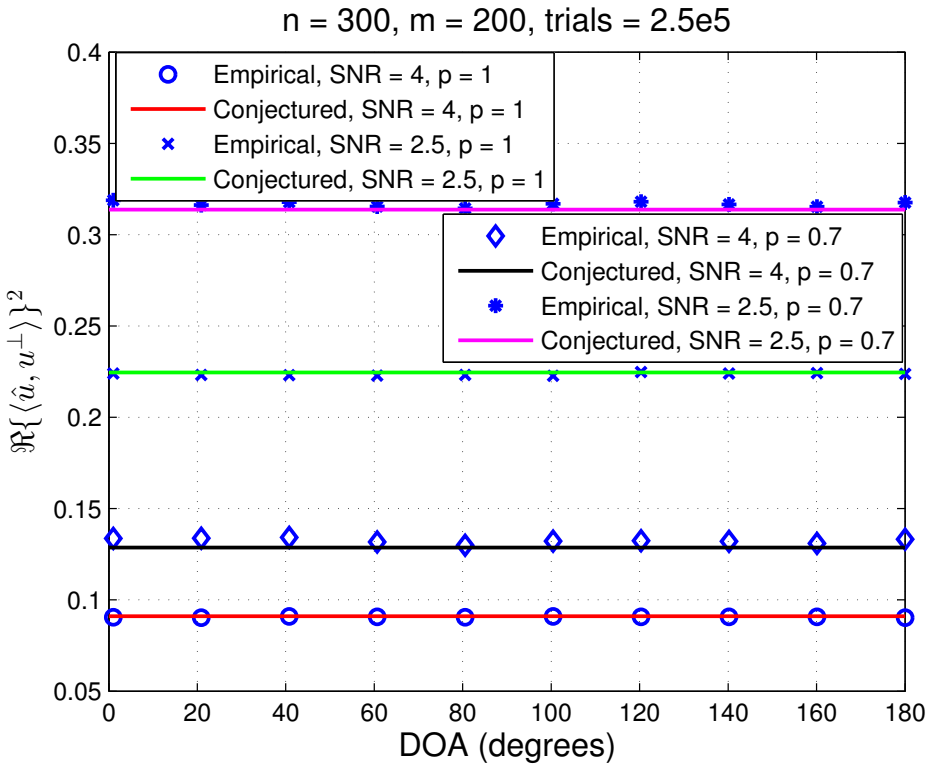


Figure 2.2: Plot of $2m\,\Re\{\widehat{u}^H q\}^2$ against directions of arrival θ, for a system with 300 sensors spaced 1/3 wavelengths apart, and 200 snapshots, where $u^{\perp}$ is given by (2.21a). The 'empirical' plots correspond to generating data via (2.4), estimating $\widehat{u}$ via the SVD of $\widehat{Y}$, and computing the average of $2m\,\Re\{\langle \hat{u}, q \rangle\}^2$ over 250000 trials. The conjectured values in each case are $1-\alpha^2$, where α^2 is computed for the different values of $\sigma = \{2.5, 5\}$ and $p = \{0.7, 1\}$ using (A.1a). We see that for the uniform linear array, irrespective of the direction of arrival, $2m\,\Re\{\widehat{u}^H q\}^2$ is accurately predicted by the conjectured value.

$$\approx |\gamma_1|^2\,\alpha^2 + \delta_1^2\,\frac{1-\alpha^2}{2m} + 2\,\delta_1\,\alpha\,\Re\{\gamma_1\,\widehat{u}^H q\} + o(1). \tag{2.30}$$

Combining (2.28) and (2.30), we obtain,

$$(\Delta\theta)_{\text{den},1} + (\Delta\theta)_{\text{den},2} \approx$$
$$\underbrace{\left(\Re\{\gamma_2\}^2 + |\gamma_1|^2\right)}_{=\,\delta_1^2 \text{ from (2.24)}}\,\alpha^2 + \delta_1^2\,\frac{1-\alpha^2}{2m} + 2\,\delta_1\,\alpha\,\Re\{\gamma_1\,\widehat{u}^H q\} + \delta_1\,\alpha\,\Re\{\widehat{u}^H q\} + o(1). \tag{2.31}$$

Now we use the results from Appendix B regarding the scaling of ULA and its derivatives with m. From (B.2), (B.6), (B.4) and (B.8) respectively, γ_1 is $O(m)$, γ_2 is $O\left(m^2\right)$, δ_1^2 is $O\left(m^2\right)$ and δ_2^2 is $O\left(m^4\right)$. This implies that the first term $\delta_1^2\alpha^2$ dominates the denominator

$(\Delta\theta)_{\text{den},1} + (\Delta\theta)_{\text{den},2}$ in (2.31). Therefore,

$$\left((\Delta\theta)_{\text{den},1} + (\Delta\theta)_{\text{den},2}\right)^2 \approx \delta_1^4\alpha^4 + o(1). \tag{2.32}$$

Subsituting (2.27) and (2.32) in (2.20) gives,

$$\mathbb{E}\left[\Delta\theta^2\right] = \frac{1-\alpha^2}{2m\delta_1^2\alpha^2} + \text{ higher order terms,}$$

so that we approximate

$$\mathbb{E}\left[\Delta\theta^2\right] \approx \frac{1-\alpha^2}{2m\delta_1^2\alpha^2}$$

which is the theoretical expression for the MSE of $\widehat{\theta}$, where α^2 is given by (A.1a), $\delta_1^2 := \delta_1(\theta)^2 = ||u^{(1)}(\theta)||_2^2 - |\gamma_1(\theta)|^2$ and $\gamma_1(\theta) = u(\theta)^H u^{(1)}(\theta)$.

This expression for the MSE highlights the role of both the array geometry and the the accuracy of subspace components estimation in the performance of the MUSIC algorithm. The term δ_1^2 in (2.16) is dependent only the geometry of the array, and the true DOA, and tells us how the error varies with true DOA location, the term $(1-\alpha^2)/(2m\alpha^2)$ tells us how the accuracy of the sample singular vector $\widehat{u}$ affects the error. This is highlighted by the plot in Figure 2.6.

II.4.1 Previous Results

Li *et al.* derived the following expression for the MSE performance of MUSIC using perturbation theory [1, p. 1178, eq. (81)].

$$\text{MSE}_{\text{LLV}} = \frac{1}{2m\sigma^2\widetilde{\gamma}^2}, \tag{2.33}$$

where $\widetilde{\gamma}^2 = ||a^{(1)}(\theta)||_2^2 - |a^{(1)}(\theta)u(\theta)|^2$, and $a^{(1)}(\theta)$ is the first derivative of the manifold vector. In their notation, $\gamma_k = \widetilde{\gamma}^2$; (2.33) may be obtained by using the definition of γ_k from Table II on p. 1178 in [1], recognizing that $\Omega_0 = I - u(\theta)u(\theta)^H$, and appropriately substituting for for σ_n, the noise variance, M, the number of sensors, and R_s, the signal covariance matrix. We see that (2.33) captures the component of MSE due to array geometry, but does not account for the accuracy of the singular vector estimates, which we gain from the random matrix theory based approach of this work. We note here that the analysis in [1] was not carried out in the large array regime. However, even for moderately sized arrays, (2.16) provides

improved accuracy in predicting the MSE at low SNRs, as shown in Figure 2.6.

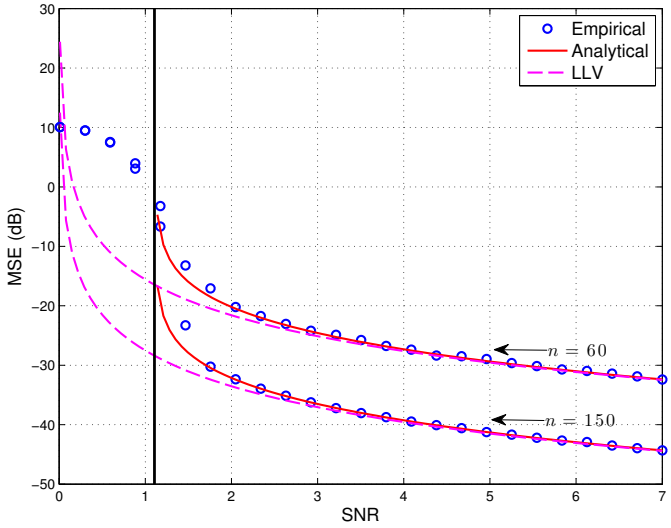
II.5 Numerical Simulations

Figure 2.3 plots the MSE of MUSIC DOA versus SNR for two systems with different number of sensors, and snapshots, both chosen so that $m/n = 1.5$. There is no missing data, i.e. $p = 1$. Figure 2.3a corresponds to the case when the sensors in the ULA are 1/2 wavelength apart, and Figure 2.3b to the case when they are 1/3 wavelength apart. This plot illustrates the validity of our analytics expression (2.16). As predicted, below critical SNR, the MUSIC algorithm breaks down completely. We also compare our predictions to the previous predictions in the literature, by Li et. al., which derived using perturbation theory in [1].

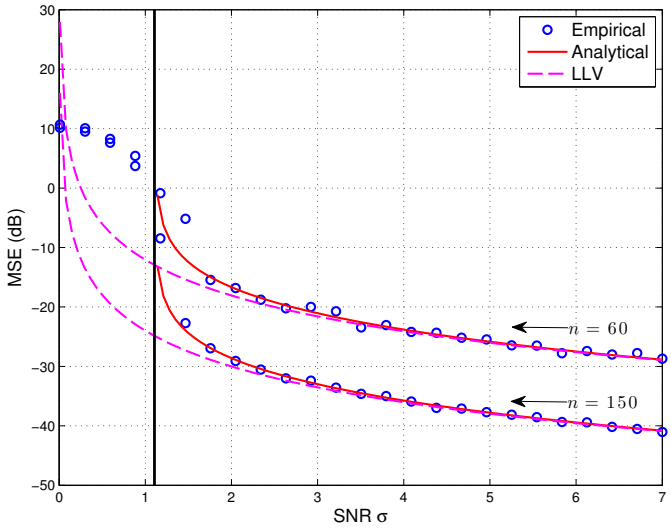
Figure 2.4 plots the MSE of DOA versus the fraction of data observed (p) a system with 150 sensors and 100 snapshots (i.e., sample deficient regime), chosen so that $m/n = 1.5$. This plot illustrates the validity of our analysis of MUSIC with missing data. At low SNR, the prediction is inaccurate close to the phase transition boundary due to the finite size of the simulated system, as opposed to the asymptotic analysis.

In Figure 2.5, we plot the heatmap of DOA MSE against varying eigen-SNR σ and fraction of data observed p. This plot highlights the validity of the phase transition boundary σ_{crit} predicted by our analysis as both σ and p are varied.

Figure 2.6 plots the DOA MSE against the true direction fo arrival θ over the entire range of DOAs, and we compare the accuracy of our predictions against the previous analysis in the literature given by [1]. The plot highlights the role of source location in DOA estimation performance, which facilitates system design.



(a) MSE of DOA estimate vs SNR, for sensors spaced 1/2 wavelength apart.



(b) MSE of DOA estimate vs SNR, for sensors spaced 1/3 wavelength apart.

Figure 2.3: Empirical MSE of DOA estimate versus SNR = σ for $m = 150$ and $n = 60$. Here we set $p = 1$, and pick the number of samples (or snapshots) $m = n/1.5$ so that $c = m/n = 1.5$ in Theorem A.1, so that we are operating in the sample deficient regime. Figure 2.3a corresponds to the case where the sensors are $d = 1/2$ wavelength apart, and Figure 2.3b corresponds to the case where the sensors are $d = 1/3$ wavelengths apart in (2.2). The theoretical prediction for the MSE was plotted by using the analytic expression in (2.16). The empirical values correspond to generating the full data matrix and running MUSIC on it; the regions where there is excellent agreement with the theoretical predictions are where the high-dimensional random matrix theoretic predictions are accurate. The predictions are less accurate as one nears the breakdown point which corresponds to values near SNR $\approx c^{1/4}/\sqrt{p}$. 'LLV predictions' corresponds to the MSE prediction (2.33), as derived in [1].

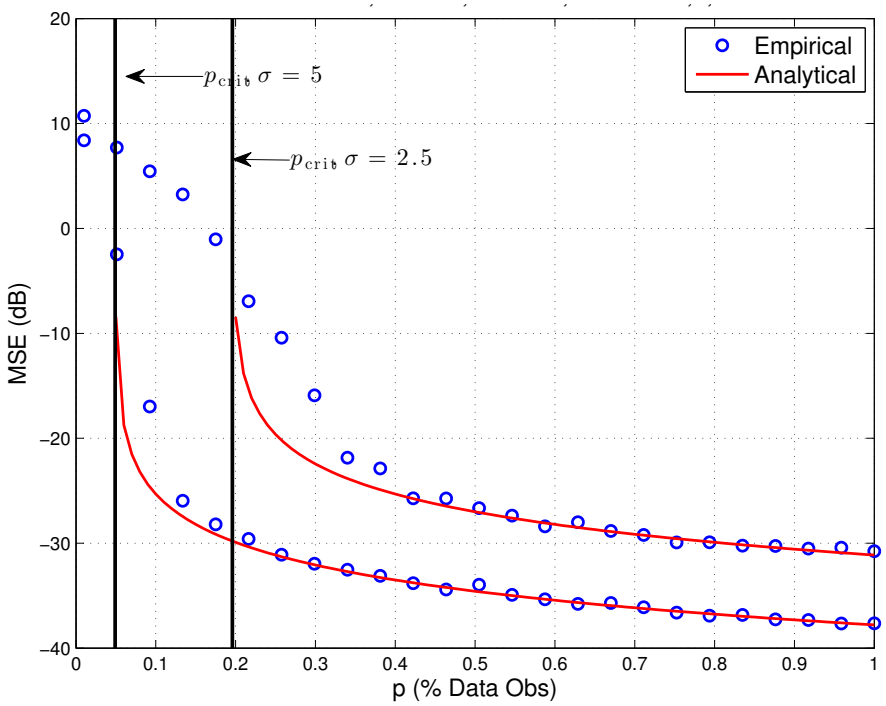


Figure 2.4: Empirical MSE of DOA estimate versus fraction of data observed = p, for two different SNRs $\sigma = 2.5$ and 5. System was moderately sized at $m = 150$ and $n = 100$ so that $c = m/n = 1.5$ in Theorem A.1. The two vertical lines correspond to theoretical breakdown points $p_{\text{crit}} = \sqrt{c}/\sigma^2$. We see that $\sigma = 5$, prediction of MSE is accurate throughout, but at low SNR, $\sigma = 2.5$, close to the phase transition region, prediction becomes inaccurate. This is due to the finite dimensions of the simulated system, while the theoretical prediction was derived for the asymptotic regime.

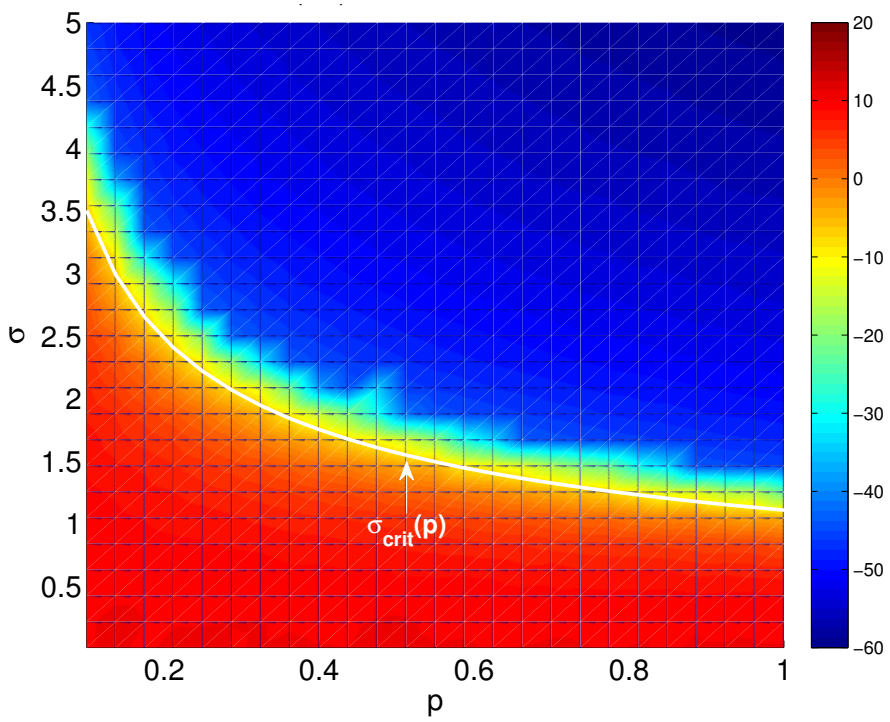


Figure 2.5: Empirical MSE of DOA estimation for a system with 600 sensors and 400 snapshots, plotted as SNR varies from $0.1 - 5$ and fraction of observed entries varies from $0.1 - 1$. The white line is a plot of the phase transition boundary σ_{crit} as a function of p. The phase transition boundary prediction is accurate for the entire range of SNR and p. Due to the finite dimensions of the system, the we see the phase transition boundary in this plot has some width, but asymptotically, it will reduce to a line with coincides with $\sigma_{\text{crit}}(p)$. Below the phase transition boundary, the theoretical MSE returned by the expression (2.16) is ∞, hence the value plotted is the variance of uniformly selecting a DOA in the range of possible values.

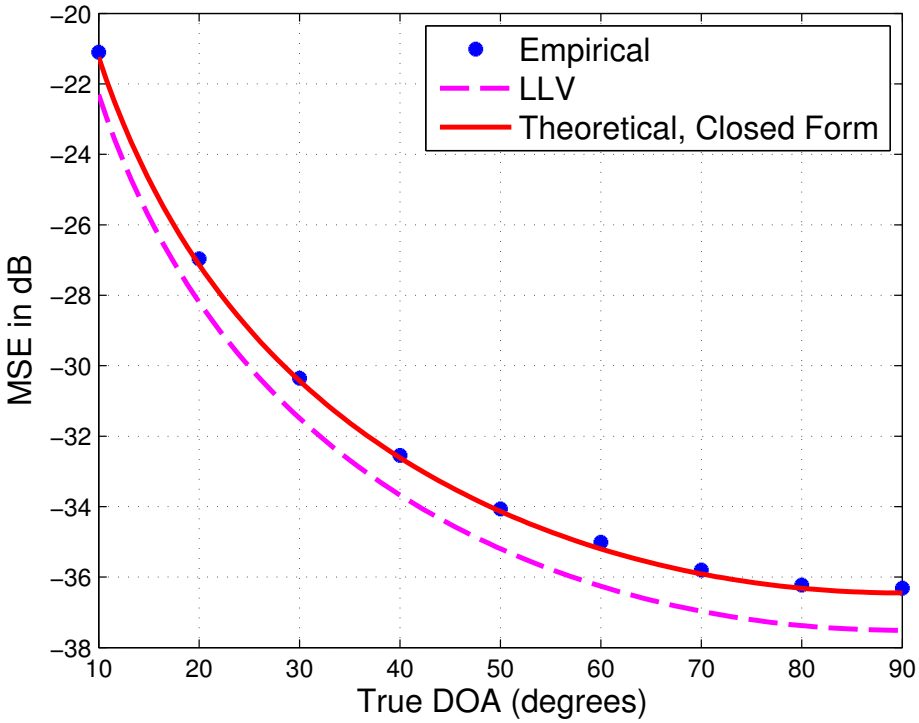


Figure 2.6: Empirical MSE of DOA estimation for a system with 150 sensors and 100 snapshots, plotted as the true DOA varies between $10^o - 90^o$. The theoretical prediction of the MSE given by (2.16) is plotted in red, and the previous prediction in literature [1] given by (2.33) is plotted as a dashed line. The ratio of observed entries is 0.8. We see that our theoretical predictions accurately predict the MSE for the entire range of DOAs, and throughout, more accurate prediction is achieved (ranging from $0.5 - 1$dB at 20^o to around 2dB at 90^o) compared to the previous result in the literature. The plot gives us insight into the role of source location and array geometry in DOA estimation performance.

II.6 Conclusions

In this chapter, we analyzed the performance of MUSIC algorithm in the white Gaussian like noise setting, with a random subset of entries observed, or missing data, for a system with a single source. Using recent results from random matrix theory, we showed that the MUSIC DOA estimate was consistent for a class of arrays we termed low coherence arrays, in the large sample, relatively large array regime, above a phase transition boundary. This phase transition region boundary depends upon the SNR, the probability of observing entries in the data matrix and the number of sensors and snapshots in a simple, explicit way and generalizes related work in the literature [39]. Furthermore, we obtained an analytic expression for the mean squared error of MUSIC DOA for a system using ULAs, and validated our analysis using simulations. Proving convergence of second order terms involving bilinear forms of random matrices, and thereby establishing Central Limit Theorem for DOA estimation remain important open problems.

CHAPTER III

Algorithms for Estimation of Rank One Signal in Noise with Double Kronecker Product Structured Singular Vectors

III.1 Introduction

We consider the problem of estimating rank one matrices buried in high dimensional noise, whose left singular vector (or right singular vector, but not both singular vectors) is assumed to have Kronecker product structure. Without loss of generality, we assume that the $m = m_1 m_2$ dimensional left singular vector of the signal is the Kronecker product of deterministic, unknown, unit norm vectors of known dimensions m_1 and m_2. Such Kronecker product structured singular vectors arise, for instance, in Space Time Adaptive Processing (STAP) [55, 16], or from appropriate rearrangement of a three dimensional $m_1 \times m_2 \times n$ tensor signal with Kruskal decomposition of rank one.

Canonically, the rank one signal matrix is estimated by the (ordered) singular value decomposition (SVD) of the observed data matrix Y truncated to rank-1. However, such estimates do not account for the Kronecker product structure of the singular vector. Let the estimates of the principal singular value, and associated left and right singular vectors of the signal, respectively, be denoted $\widehat{\sigma}$, $\widehat{u}$ and $\widehat{s}$. We focus on the computation of $\widehat{u}$, by exploiting the Kronecker product structure underlying it. Once $\widehat{u}$ has been computed, the unknown rank one matrix can be inferred from $\widehat{u}$ via the relation

$$||\widehat{u}^H Y||_2 \cdot \widehat{u} \cdot \frac{\widehat{u}^H Y}{||\widehat{u}^H Y||_2}.$$

For the remainder of the chapter, we will focus on algorithms for estimating $\widehat{u}$ and their relative performance with the underlying assumption that improved estimation of $\widehat{u}$ will improve the estimation of the rank one signal matrix.

In related previous work, Tsiligkaridis and Hero consider the estimation of covariance matrices which could be decomposed as the sum of Kronecker products of smaller dimensional matrices [56]. Their algorithm utilized a rearrangement operator introduced by Van Loan and Pistianis [19], to first rearrange the sample covariance matrix to exploit the Kronecker product structure, and thereby estimate the underlying Kronecker components. Greenwald and Hero extended this work to estimate structured Kronecker components – specifically when one of the Kronecker components is a Toeplitz matrix [57]. They also introduced an iterative algorithm when the Kronecker component matrices have low rank [2], with applications to STAP. Ritsey et. al. also exploited the Kronecker structure for signal estimation via alternating least squares, and consequent beamforming for STAP [58]. In this chapter, we consider a signal-plus-noise-model where the latent rank one signal matrix has a left singular vector which is a Kronecker product of two unknown vectors. We formulate our problem to estimate the underlying rank one signal directly from the rectangular data matrix, as opposed to the square covariance matrix as done in [56, 57, 2].

In Section III.2, we show that our data model (3.1) may be obtained from a rearrangement of a an equivalent tensor valued signal-plus-noise model. Thus, an alternate approach to the estimation of the underlying rank one matrix is to rearrange the data appropriately into a tensor and estimate the underlying rank one tensor.The problem we address in this work is equivalent to estimating tensor valued signals of order three and rank one. The estimation of rank one tensors is also of special interest in the literature, since it is a central step in the estimation of higher order tensors through deflation based techniques [59, 60, 61].

The new contributions of this chapter are as follows: (a) We propose four algorithms for estimating the Kronecker structured singular vector that utilizes the Van Loan-Pitsianis rearrangement operator in different ways. These algorithms add to the variety of algorithms which are available in the literature for the estimation of Kronecker structured rank one signals. (b) Using matrix perturbation theory, we prove that all four algorithms are stable to additive noise, i.e. that the estimates returned by the algorithms converge to the true underlying singular vectors as the noise reduces to zero. (c) We provide an analysis of the relative estimation performance of all the algorithms that builds on insights from random matrix theory. These analyses highlights new aspects of algorithm performance (their stability), and the use of new tools for their characterization (using random matrix theory), which

complements other rank one signal/tensor estimation algorithms in the literature, such as [62, Prop. 4.1] alluded to above.

Furthermore, our analysis brings into sharper focus a phase transition boundary, which separates regimes where the algorithms return accurate estimates of the underlying singular vectors and where the algorithms fail. We validate the resulting theoretical predictions using extensive numerical simulations. We compare the estimation performance of the algorithms proposed in this chapter to previous approaches in the literature, and we show improved clutter suppression in the STAP application using the algorithms proposed in this chapter relative to the canonical SVD estimate.

Our simulations and performance analyses reveal that the best performance is achieved by the algorithm which first computes the SVD estimates of the singular vectors, and then utilizes the Kronecker structure to effectively denoise the estimate. Rearranging the data first to exploit the Kronecker structure results in a performance loss.

The development of the algorithms and their stability is the primary focus of this chapter, which is organized as follows. We describe the signal-plus-noise model in Section III.2, the Van Loan-Pitsianis rearrangement operator in Section III.3, and the four new algorithms in Section III.4. We prove the stability of the proposed algorithms in Section III.5. We analyze the relative estimation performance of the algorithms in Section III.6. Numerical simulations are presented in Section III.7 and concluding remarks in Section III.8.

III.2 Model and Estimation Problem

We consider the signal-plus-noise model

$$Y = \sigma_0 \underbrace{(a \otimes b)}_{:=u} s^H + Z \tag{3.1}$$

where $Y \in \mathbb{C}^{m \times n}$ is the observed data matrix, $\sigma_0 \in \mathbb{R}_+$ is the signal-to-noise ratio (SNR), and $a \in \mathbb{C}^{m_1 \times 1}$, $b \in \mathbb{C}^{m_2 \times 1}$ (and hence $u \in \mathbb{C}^m$) are unknown, non-random unit norm vectors. Here $\otimes$ denotes the Kronecker product [18, Chapter 13]. The signal vector s is unit norm without loss of generality, and Z is the noise-only matrix with i.i.d. zero mean, variance $1/n$ entries with bounded higher order moments.

Let

$$Y = \widehat{\sigma}\,\widehat{u}\,\widehat{v}^H \tag{3.2}$$

be the (ordered) SVD of Y truncated to rank one. Then $\widehat{u}$, the principal left singular vector of Y, is the canonical estimate of u in (3.1). However, $\widehat{u}$ does not account for the Kronecker product structure of u. Our goal is to propose algorithms for estimating u in a way which accounts for its Kronecker product structure: we expect such algorithms to return more accurate estimates of u.

III.2.1 Connections to Low Rank Tensor Estimation

The problem of estimating u is intimately linked to the problem of low rank tensor estimation. Details about tensors, their algebra, and their estimation may be found in [63, 60, 64] *etc.* Here we note only the details relevant to our problem.

Let

$$\mathcal{Y} = \sigma_0\,\bar{s} \circ b \circ a + \mathcal{Z}$$

where $\circ$ denotes outerproduct, and $\mathcal{Y}, \mathcal{Z} \in \mathbb{C}^{n \times m_1 \times m_2}$ are tensors of order four. Let $\mathcal{Y}_{[j]}$ denote *unfolding* $\mathcal{Y}$ along mode j. The unfolding operation extracts fibers from mode j of $\mathcal{Y}$, and arranges them as columns of a matrix. Unfolding $\mathcal{Y}$ along mode 1,

$$\mathcal{Y}_{[1]} = \sigma_0\,\bar{s}\,(a \otimes b)^T + \mathcal{Z}_{[1]}. \tag{3.3}$$

Comparing (3.3) with (3.1), we see that $\mathcal{Y}_{[1]} = Y^T$. Low rank estimation of Y is thus equivalent to the low rank estimation of $\mathcal{Y}$.

III.3 Van Loan-Pitsianis Rearrangement Operator

In this section, we describe the rearrangement operation introduced by Van Loan and Pitsianis [19]. The operation will be central to the description of our new algorithms in Section III.4.

Suppose

$$U = A \otimes B$$

where $U \in \mathbb{C}^{m_1 m_2 \times n_1 n_2}$, $A \in \mathbb{C}^{m_1 \times n_1}$ and $B \in \mathbb{C}^{m_2 \times n_2}$. Partition U into block submatrices

$U_{ij} \in \mathbb{C}^{m_2 \times n_2}$, $i, j \in \{1, \ldots, m_1\}, \{1, \ldots, n_1\}$ such that

$$U = \begin{bmatrix} U_{11} & \ldots & U_{1,n_1} \\ & \ddots & \\ U_{m_1,1} & \ldots & U_{m_1,n_1} \end{bmatrix}.$$

Let $\mathcal{R}(U) : \mathbb{C}^{m_1 m_2 \times n_1 n_2} \rightarrow \mathbb{C}^{m_1 n_1 \times m_2 n_2}$ be a rearrangement operation, whose input is U and output is:

$$\mathcal{R}(U) = \begin{bmatrix} U_1 & \ldots & U_{n_1} \end{bmatrix}^T \tag{3.4}$$

where, for $j = 1, \ldots, n_1$:

$$A_j = \begin{bmatrix} \mathrm{vec}\left(A_{1,j}\right)^T \\ \vdots \\ \mathrm{vec}\left(A_{m_1,j}\right)^T \end{bmatrix}.$$

Here $\mathrm{vec}(\cdot)$ vectorizes the input matrix by stacking its columns into one vector:

$$X \in \mathbb{C}^{m \times n} \implies \mathrm{vec}(X) = \begin{bmatrix} X(1:m,1) \\ X(1:m,2) \\ \vdots \\ X(1:m,n) \end{bmatrix} \in \mathbb{C}^{mn} \tag{3.5}$$

and let reshape$(\cdot)$ be the inverse operation such that

$$\mathrm{reshape}_{mn \times 1 \rightarrow m \times n}(\mathrm{vec}(X)) = X.$$

The utility of $\mathcal{R}(\cdot)$ is that it transforms a Kronecker product decomposition problem into a low rank decomposition problem:

$$\|A - B \otimes C\|_F = \|\mathcal{R}(A) - \mathrm{vec}(B)\,\mathrm{vec}(C)^T\|_F \tag{3.6}$$

and the solution to the latter problem is known through the well known Eckart-Young-Mirksy theorem [65].

III.4 New Algorithms for the Estimation of u

In this Section, we present four new algorithms, KPCA2,1 KPCA2,2ℓ, KPCA2,2r, and KPCA2,3, (collectively, "KPCA2") for the estimation of u from (3.1).

Note. Throughout the rest of this chapter, x^* denotes the complex conjugate of x.

Algorithm 3.1: KPCA2,1

1 $\widehat{u}_{\mathsf{svd}}$ = principal left singular vector of Y /* 1st SVD */

2 $\widehat{U}_{\mathcal{R}}$ = $\mathcal{R}_{m_1\, m_2\times 1\to m_1\times m_2}(\widehat{u}_{\mathsf{svd}})$

3 $\widehat{a}_{\mathsf{kpca2,1}}$ = principal left singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

4 $\widehat{b}^*_{\mathsf{kpca2,1}}$ = principal right singular vector of $\widehat{U}_{\mathcal{R}}$

Algorithm 3.2: KPCA2,2ℓ

1 $Y_{\mathcal{R}}$ = $\mathcal{R}_{m_1\, m_2\times n\to m_1\times m_2\, n}(Y)$

2 $\widehat{a}_{\mathsf{kpca2,2\ell}}$ = principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

3 $\widehat{v}_1^*$ = principal right singular vector of $Y_{\mathcal{R}}$

4 $\widehat{V}_1$ = reshape$_{m_2\, n\times 1\to m_2\times n}(\widehat{v}_1)$

5 $\widehat{b}_{\mathsf{kpca2,2\ell}}$ = principal left singular vector of $\widehat{V}_1$ /* 2nd SVD */

Algorithm 3.3: KPCA2,2r

1 $Y_{\mathcal{R}}$ = $\mathcal{R}_{m_1\, m_2\times n\to m_1\, n\times m_2}(Y)$

2 $\widehat{b}^*_{\mathsf{kpca2,2r}}$ = principal right singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

3 $\widehat{u}_1$ = principal left singular vector of $Y_{\mathcal{R}}$

4 $\widehat{U}_1$ = reshape$_{m_1\, n\times 1\to m_1\times n}(\widehat{u}_1)$

5 $\widehat{a}_{\mathsf{kpca2,2r}}$ = principal left singular vector of $\widehat{U}_1$ /* 2nd SVD */

Algorithm 3.4: KPCA2,3

1 $Y_{\mathcal{R}} = \mathcal{R}_{m_1\, m_2 \times n \to m_1 \times m_2\, n}(Y)$

2 $\widehat{a}_{\mathsf{kpca2,3}}$ = principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

3 $Y_{\mathcal{R}'} = \mathcal{R}_{m_1\, m_2 \times n \to m_1\, n \times m_2}(Y)$

4 $\widehat{b}^{*}_{\mathsf{kpca2,3}}$ = principal right singular vector of $Y_{\mathcal{R}'}$ /* 2nd SVD */

The algorithms are outlined in Algorithms 3.1, 3.2, 3.3, 3.4. Each algorithm uses the rearrangement operation from Section III.3 in a different way, and involves two SVD computations. Each algorithm gives us $\widehat{a}, \widehat{b}$, estimates of a, b. From these, we may compute a new estimate of u as

$$\widehat{u} = \widehat{a} \otimes \widehat{b}. \tag{3.7}$$

We provide the detailed derivation of the algorithms below.

III.4.1 KPCA2,1

This algorithm first estimates u through the SVD of the data, and then fits a Kronecker product model to the SVD based estimate.

In Step 1, we compute $\widehat{u}_{\mathsf{svd}}$, the SVD based estimate of u. Since u has a double Kronecker product structure, we solve the optimization problem

$$\widehat{a}_{\mathsf{kpca2,1}}, \widehat{b}_{\mathsf{kpca2,1}} = \underset{\substack{a,b: \\ \|a\|_2=1, \|b\|_2=1}}{\arg\min} \|\widehat{u}_{\mathsf{svd}} - a \otimes b\|_F^2. \tag{3.8}$$

In Step 2, we apply the rearrangement defined in (3.4):

$$\widehat{U}_{\mathcal{R}} := \mathcal{R}_{m_1\, m_2 \to m_1 \times m_2}(\widehat{u}_{\mathsf{svd}})$$

so that, as in (3.6),

$$\|\widehat{u}_{\mathsf{svd}} - a \otimes b\|_F^2 = \|\widehat{U}_{\mathcal{R}} - \mathrm{vec}\,(a)\,\mathrm{vec}\,(b)^T\,\|_F^2 = \|\widehat{U}_{\mathcal{R}} - ab^T\|_F^2,$$

where we have used the fact that $a = \mathrm{vec}\,(a)$, $b = \mathrm{vec}\,(b)$ since they are vectors. Therefore,

the problem (3.8) is equivalent to

$$\widehat{a}_{\mathsf{kpca2,1}},\, \widehat{b}_{\mathsf{kpca2,1}} = \underset{\substack{a,b:\\ \|a\|_2=1,\|b\|_2=1}}{\arg\min} \quad \|\widehat{U}_{\mathcal{R}} - ab^T\|_F^2. \tag{3.9}$$

In Step 3 and 4, we compute the solution to (3.9) through the EYM theorem [65],

$$\widehat{a}_{\mathsf{kpca2,1}} = \text{principal left singular vector of } \widehat{U}_{\mathcal{R}}, \tag{3.10a}$$

$$\widehat{b}^*_{\mathsf{kpca2,1}} = \text{principal right singular vector of } \widehat{U}_{\mathcal{R}}. \tag{3.10b}$$

Given estimates $\widehat{a}_{\mathsf{kpca2,1}}$, $\widehat{b}_{\mathsf{kpca2,1}}$, as in (3.10a), (3.10a) we compute $\widehat{u}_{\mathsf{kpca2,1}}$ using (3.7).

III.4.2 KPCA2,2ℓ

In this algorithm, we first rearrange the data matrix to exploit the Kronecker product structure, and then compute its SVD.

In Step 1, we apply the rearrangement

$$Y_{\mathcal{R}} := \mathcal{R}_{m_1\, m_2\times n\to m_1\times m_2\, n}(Y) = \sigma_0\, a\, \mathrm{vec}\left(bs^H\right)^T + \mathcal{R}(Z) \tag{3.11}$$

where $\mathcal{R}(\cdot)$, $\mathrm{vec}\,(\cdot)$, are defined in (3.4), (3.5) respectively, and we have used the fact that $\mathrm{vec}\,(a) = a$. We have that:

$$\underset{\substack{a,b,s,\sigma:\\ \|a\|_2=1,\|b\|_2=1,\|s\|_2=1}}{\arg\min} \quad \|Y - \sigma\,(a\otimes b)\,s^H\|_F^2 = \underset{\substack{a,b,s,\sigma:\\ \|a\|_2=1,\|b\|_2=1,\|s\|_2=1}}{\arg\min} \quad \|Y_{\mathcal{R}} - \sigma\, a\,\mathrm{vec}(b\,s^H)^T\|_F^2. \tag{3.12}$$

In Step 2, obtain the following estimate of a, suggested by (3.12):

$$\widehat{a}_{\mathsf{kpca2,2}\ell} = \text{principal left singular vector of } Y_{\mathcal{R}}. \tag{3.13a}$$

In Step 3, we see that $\widehat{v}_1$, the principal right singular vector of $Y_{\mathcal{R}}$, is an estimate of $\mathrm{vec}\left(bs^H\right)$. Hence, in Step 4, we apply the reshaping

$$\widehat{V}_1 = \mathrm{reshape}_{m_2\, n\times 1\to m_2\times n}(\widehat{v}_1)$$

so that $\widehat{V}_1$ is an estimate of bs^H. We find an estimate of b by solving the optimization problem

$$\underset{\substack{b,s:\\ \|b\|_2=1,\|s\|_2=1}}{\arg\min} \quad \|\widehat{V}_1 - bs^H\|_F^2.$$

Thus, in Step 5, we obtain the following estimate of b:

$$\widehat{b}_{\mathsf{kpca2},2\ell} = \text{principal left singular vector of } \widehat{V}_1. \tag{3.13b}$$

Given $\widehat{a}_{\mathsf{kpca2},2\ell}$, $\widehat{b}_{\mathsf{kpca2},2\ell}$ from (3.13a), (3.13b), we can compute $\widehat{u}_{\mathsf{kpca2},2\ell}$ using (3.7).

III.4.3 KPCA2,2r

This algorithm is similar to KPCA2,2ℓ of Section III.4.2, except that we associate s with a — instead of b as in (3.11) — while rearranging the data. In Step 1, we apply the rearrangement:

$$Y_{\mathcal{R}} := \mathcal{R}_{m_1\, m_2\times n\to m_1\, n\times m_2}(Y) = \sigma_0 \operatorname{vec}\left(as^H\right) b^T + \mathcal{R}(Z). \tag{3.14}$$

Following steps analogous to equations (3.11) through (3.13a), we have that:

$$\widehat{b}^*_{\mathsf{kpca2},2\mathsf{r}} = \text{principal right singular vector of } Y_{\mathcal{R}} \tag{3.15a}$$

and that $\widehat{u}_1$, the principal left singular vector of $Y_{\mathcal{R}}$, is an estimate of $\operatorname{vec}\left(as^H\right)$. Using the rearrangement

$$\widehat{U}_1 = \operatorname{reshape}_{m_1\, n\times 1\to m_1\times n}(\widehat{u}_1),$$

and following steps analogous to (3.13a) through (3.13b), we have that

$$\widehat{a}_{\mathsf{kpca2},2\mathsf{r}} = \text{principal left singular vector of } \widehat{U}_1. \tag{3.15b}$$

Given $\widehat{a}_{\mathsf{kpca2},2}$, $\widehat{b}_{\mathsf{kpca2},2}$ from (3.15b), (3.15a), we compute $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$ using (3.7).

III.4.4 KPCA2,3

This algorithm combines the ideas of KPCA2,2ℓ and KPCA2,2r. In those algorithms, we exploit the Kronecker product structure of the singular vector directly from the data, to estimate one of the Kronecker product components of u: a in KPCA2,2ℓ and b in KPCA2,2r.

Here, we apply the rearrangement operator to the data twice — first to estimate a as in KPCA2,2ℓ, and then to estimate b as in KPCA2,2r. Therefore,

$$\widehat{a}_{\mathsf{kpca2,3}} = \widehat{a}_{\mathsf{kpca2,2}\ell} \text{ as in (3.13a)}, \tag{3.16a}$$

$$\widehat{b}_{\mathsf{kpca2,3}} = \widehat{b}_{\mathsf{kpca2,2r}} \text{ as in (3.15a)}. \tag{3.16b}$$

Given estimates $\widehat{a}_{\mathsf{kpca2,3}}$, $\widehat{b}_{\mathsf{kpca2,3}}$, as in (3.16a), (3.16a) we compute $\widehat{u}_{\mathsf{kpca2,1}}$ using (3.7).

III.5 Algorithms Stability via Matrix Perturbation Theory

In this section, we show that the KPCA algorithms are stable, i.e. $|\langle \widehat{u}, u\rangle|^2 \to 1$ as $\|Z\|_F \to 0$ in (3.1), where $\langle\cdot,\cdot\rangle$ denotes the inner product, and $\widehat{u}$ is the estimate returned by any of the KPCA2 algorithms. The analysis draws upon the following Proposition.

Proposition III.1 (Wedin)**.** *Suppose we have matrices $A, \widehat{A}$ and Z such that $\widehat{A} = A + Z$. Let the SVD of $\widehat{A}$ and A, respectively, be partitioned as follows:*

$$\widehat{A} = \begin{bmatrix} \widehat{U}_1 & \widehat{U}_2 \end{bmatrix} \begin{bmatrix} \widehat{\Sigma}_1 & 0 \\ 0 & \widehat{\Sigma}_2 \end{bmatrix} \begin{bmatrix} \widehat{V}_1^H \\ \widehat{V}_2^H \end{bmatrix}, \tag{3.17a}$$

$$A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix}. \tag{3.17b}$$

If there exists $\delta > 0$, such that

$$\alpha + \delta \le \min\{\widehat{\sigma}_1, \ldots, \widehat{\sigma}_r\} \quad \textit{and} \quad \alpha \ge \max\{\sigma_{r+1}, \ldots, \sigma_n\}, \tag{3.18}$$

then we have that

$$\max\{\|\sin\Theta\|_F, \|\sin\Phi\|_F\} \le \frac{\max\left\{\|(\widehat{A} - A)^H U_1\|_F,\ \|(\widehat{A} - A) V_1\|_F\right\}}{\delta} \tag{3.19}$$

for any $\delta \in (0, \sigma_1(X)]$.

Here, Θ and Φ are respectively, the canonical angles between the subspaces $\{U_1, \widehat{U}_1\}$ and

$\{V_1, \widehat{V}_1\}$, *defined as*

$$\sin\Theta := \operatorname{diag}\left(\sigma_1(U_2^H \widehat{U}_1),\ \sigma_2(U_2^H \widehat{U}_1),\ \dots\right), \tag{3.20a}$$

$$\sin\Phi := \operatorname{diag}\left(\sigma_1(V_2^H \widehat{V}_1),\ \sigma_2(V_2^H \widehat{V}_1),\ \dots\right), \tag{3.20b}$$

where $\sigma_k(\cdot)$ *denotes the* k^{th} *largest singular value of its argument.*

Proof. See [66]. □

Corollary III.1. *Suppose* $Y = \sigma_0\, u\, s^H + Z$, *where* u, s *are unit norm vectors. Let* $\widehat{u}_{\mathsf{svd}}$ *and* $\widehat{v}_{\mathsf{svd}}$ *be the left and right singular vectors of* Y *corresponding to its largest singular value,* $\widehat{\sigma}_1$. *Then,*

$$\max\{1 - |\langle \widehat{u}_{\mathsf{svd}}, u\rangle|^2, 1 - |\langle \widehat{v}_{\mathsf{svd}}, u\rangle|^2\} \le \frac{\|Z\|_F^2}{\delta_1^2}, \tag{3.21}$$

for any $\delta_1 \in (0, \widehat{\sigma}_1]$.

Proof. Using the definition (3.20),

$$\|\sin\Theta\|_F^2 = \|\widehat{U}_2^H u\|_F^2 = u^H \left(I - \widehat{u}_{\mathsf{svd}}\, \widehat{u}_{\mathsf{svd}}^H\right) u = 1 - |\langle \widehat{u}_{\mathsf{svd}}, u\rangle|^2. \tag{3.22a}$$

Similarly,

$$\|\sin\Phi\|_F^2 = 1 - |\langle \widehat{v}_{\mathsf{svd}}, v\rangle|^2. \tag{3.22b}$$

Furthermore, since Frobenius norm is a consistent norm, and u, s are unit norm,

$$\|(X - S)\, s\|_F \le \ \|Z\|_F\, \|s\|_2 = \|Z\|_F, \tag{3.23a}$$

$$\|(X - S)^H\, u\|_F \le \|Z^H\|_F\, \|u\|_2 = \|Z\|_F. \tag{3.23b}$$

Since the signal matrix is rank-1,

$$\Sigma_2 = 0 \implies \max\{\sigma_2, \dots, \sigma_n\} = 0.$$

Therefore, we can set $\alpha = 0$ in (3.18), and (3.19) holds for any $\delta \in (0, \widehat{\sigma}_1]$. Substituting (3.22) and (3.23) in (3.19) gives us (3.21), the required result. □

We now establish similar bounds on the estimation performance of the KPCA algorithms.

Theorem III.1. *Suppose we are in the setting of Corollary III.1. Let* $\widehat{u}_{\mathsf{kpca2,1}}$, $\widehat{u}_{\mathsf{kpca2,2}\ell}$, $\widehat{u}_{\mathsf{kpca2,2r}}$, $\widehat{u}_{\mathsf{kpca2,3}}$ *be computed as described by Algorithms 3.1, 3.2, 3.3, 3.4 respectively. Then,*

$$
\begin{aligned}
|\langle \widehat{u}_{\mathsf{kpca2,1}}, u\rangle|^2 &\to 1,\\
|\langle \widehat{u}_{\mathsf{kpca2,2}\ell}, u\rangle|^2 &\to 1,\\
|\langle \widehat{u}_{\mathsf{kpca2,2r}}, u\rangle|^2 &\to 1,\\
|\langle \widehat{u}_{\mathsf{kpca2,3}}, u\rangle|^2 &\to 1,
\end{aligned}
$$

as $||Z||_F \to 0$.

Proof. The outline of the proof is as follows.

1. We use Wedin's Theorem (Proposition III.1) to characterize the principal singular vectors resulting from the first SVD.
2. We express the sample singular vector as true singular vector + a residual term related to the angle between the estimated and true singular vectors.
3. We use Proposition III.1 once again to characterize the principal singular vectors resulting from the second SVD.
4. From Steps 1–3, we show that $|\langle \widehat{a}, a\rangle|^2 \to 1$ and $|\langle \widehat{b}, b\rangle|^2 \to 1$, where $\widehat{a}$, $\widehat{b}$ are the estimates of a, b returned by KPCA2. Hence, for each algorithm, we show that as $||Z||_F \to 0$
$$
|\langle \widehat{u}, u\rangle|^2 = |(\widehat{a}\otimes\widehat{b})^H (a\otimes b)|^2 = |\langle \widehat{a}, a\rangle|^2 \, |\langle \widehat{b}, b\rangle|^2 \to 1. \tag{3.25}
$$

We now consider each algorithm individually.

III.5.1 Stability of KPCA2,1

In Step 1, we compute $\widehat{u}_{\mathsf{svd}}$ from Y. The angle between the subspaces span$(\widehat{u}_{\mathsf{svd}})$ and span(u) is characterized by (3.21):

$$
\widehat{u}_{\mathsf{svd}} = \begin{bmatrix} u & U_2 \end{bmatrix} \begin{bmatrix} u & U_2 \end{bmatrix}^H \widehat{u}_{\mathsf{svd}} = u\,(u^H \widehat{u}_{\mathsf{svd}}) + U_2\,(U_2^H \widehat{u}_{\mathsf{svd}}). \tag{3.26}
$$

where U_2 is a basis of the subspace orthogonal to span(u). In Step 2, we compute $\mathcal{R}(\widehat{u}_{\mathsf{svd}})$. From (3.26),

$$\mathcal{R}(\widehat{u}_{\mathsf{svd}}) = (u^H \widehat{u}_{\mathsf{svd}}) \underbrace{a\, b^H}_{=\mathcal{R}(u)} + \mathcal{R}\left(U_2\, U_2^H \widehat{u}_{\mathsf{svd}}\right). \tag{3.27}$$

Equation (3.27) describes a rank one signal plus noise model. Let

$$u^H \widehat{u}_{\mathsf{svd}} = r e^{j\phi}.$$

The principal left and right singular vectors of $(u^H \widehat{u}_{\mathsf{svd}})\, ab^H$ are $a\, e^{j\phi}$ and b, and of $\mathcal{R}(\widehat{u}_{\mathsf{svd}})$ are $\widehat{a}_{\mathsf{kpca2,1}}$ and $\widehat{b}_{\mathsf{kpca2,1}}$ respectively. By Corollary III.1, for $\delta_2 \in (0, \sigma_1\,(\mathcal{R}(\widehat{u}_{\mathsf{svd}}))]$,

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2,1}}, a\, e^{j\phi}\rangle|^2, 1 - |\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2\} \leq \frac{||\mathcal{R}\left(U_2\, U_2^H \widehat{u}_{\mathsf{svd}}\right)||_F^2}{\delta_2^2}.$$

We may ignore $e^{j\phi}$ due to the absolute value. Furthermore, since Frobenius norm of a matrix is invariant to a rearrangement of its elements,

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2, |\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2\} \leq \frac{||U_2\, U_2^H \widehat{u}_{\mathsf{svd}}||_F^2}{\delta_2^2}. \tag{3.28}$$

$$||U_2\, U_2^H \widehat{u}_{\mathsf{svd}}||_F^2 = \widehat{u}_{\mathsf{svd}}^H\, U_2\, U_2^H\, U_2\, U_2^H\, \widehat{u}_{\mathsf{svd}}.$$

Now, since $U_2 U_2^H$ is a projection matrix and hence idempotent,

$$\begin{aligned} ||U_2\, U_2^H \widehat{u}_{\mathsf{svd}}||_F^2 &= \widehat{u}_{\mathsf{svd}}^H\, U_2\, U_2^H\, U_2\, U_2^H\, \widehat{u}_{\mathsf{svd}} \\ &= \widehat{u}_{\mathsf{svd}}^H\, U_2\, U_2^H\, \widehat{u}_{\mathsf{svd}} \\ &= \widehat{u}_{\mathsf{svd}}^H (I - u\, u^H) \widehat{u}_{\mathsf{svd}} \\ &= ||\sin\Theta||_F^2 \end{aligned} \tag{3.29}$$

from (3.22a). Combining (3.29) and (3.21), we have that for all $\delta_1 \in (0, \widehat{\sigma}]$, $\delta_2 \in (0, \sigma_1(\mathcal{R}(\widehat{u}_{\mathsf{svd}}))]$:

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2, 1 - |\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2\} \leq \frac{||Z||_F^2}{\delta_1^2\, \delta_2^2}. \tag{3.30}$$

From (3.30), we see that as $||Z||_F \to 0$, $|\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2 \to 1$, $|\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2 \to 1$, and from (3.25), that $|\langle \widehat{u}_{\mathsf{kpca2,1}}, u\rangle|^2 \to 1$ which is the required result.

III.5.2 Stability of KPCA2,2ℓ

The rearrangement of Y in Step 1 gives us:

$$Y_{\mathcal{R}} = \sigma_0\, a\, \mathrm{vec}(bs^H) + \mathcal{R}(Z).$$

Note that $Y_{\mathcal{R}}$ has a rank one signal plus noise structure. Its principal left and right singular vectors, $\widehat{a}_{\mathsf{kpca2,2}\ell}$ and $\widehat{v}$, are characterized by Corollary III.1. For any $\delta_{2\ell,1} \in (0, \sigma_1(\mathcal{R}(X))]$:

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2,2}\ell}, a\rangle|^2, 1 - |\langle \widehat{v}, \mathrm{vec}(bs^H)\rangle|^2\} \le \frac{||\mathcal{R}(Z)||_F^2}{\delta_{2\ell,1}^2}. \tag{3.31}$$

Let $v_1 := \mathrm{vec}(bs^H)$ and V_2 be an orthonormal basis for the orthocomplement of v_1. Then

$$\begin{aligned} \widehat{v}_1 &= \begin{bmatrix} v_1 & V_2 \end{bmatrix} \begin{bmatrix} v_1 & V_2 \end{bmatrix}^H \widehat{v}_1 \\ &= (v_1^H \widehat{v}_1)\, v_1 + V_2\, (V_2^H \widehat{v}_1). \end{aligned} \tag{3.32}$$

Then, due to reshape$(\widehat{v}_1)$ in Step 4:

$$\widehat{V}_1 := \mathrm{reshape}(\widehat{v}_1) = (v_1^H \widehat{v}_1)\, bs^H + \mathrm{reshape}\left(V_2\, V_2^H\, \widehat{v}_1\right). \tag{3.33}$$

Let

$$v_1^H \widehat{v}_1 = r\, e^{j\phi}. \tag{3.34}$$

The principal left and right singular vectors of $(v_1^H \widehat{v}_1) bs^H$ are $e^{j\phi} b$ and s and of $\widehat{V}$ are $\widehat{b}_{\mathsf{kpca2,2}\ell}$ and $\widehat{s}$ respectively. By Corollary III.1, for all $\delta_{2\ell,2} \in (0, \sigma_1(\mathrm{reshape}(\widehat{v}))]$,

$$\max\{1 - |\langle \widehat{b}_{\mathsf{kpca2,2}\ell}, b\, e^{j\phi}\rangle|^2, 1 - |\langle \widehat{s}, s\rangle|^2\} \le \frac{||\mathrm{reshape}\left(V_2\, V_2^H\, \widehat{v}\right)||_F^2}{\delta_{2\ell,2}^2}. \tag{3.35}$$

Using similar computations from (3.28) through (3.30), with $\widehat{a}_{\mathsf{kpca2,1}} = \widehat{b}_{\mathsf{kpca2,2}\ell}$, $\widehat{b}_{\mathsf{kpca2,1}} = \widehat{s}$, $U_2 = V_2$, and $\mathcal{R}(U_2\, U_2^H\, \widehat{u}_{\mathsf{svd}}) = \mathrm{reshape}(V_2\, V_2^H\, \widehat{v})$, we see that for every $\delta_1 \in (0, \sigma_1(\mathcal{R}(X))]$, $\delta_2 \in (0, \sigma_1(\mathcal{R}(\widehat{v}))]$:

$$\max\{1 - |\langle \widehat{b}_{\mathsf{kpca2,2}\ell}, b\rangle|^2, 1 - |\langle \widehat{s}, s\rangle|^2\} \le \frac{||Z||_F^2}{\delta_{2\ell,1}^2\, \delta_{2\ell,2}^2}. \tag{3.36}$$

From (3.31) and (3.36), we see that as $||Z||_F \to 0$, $|\langle \widehat{a}_{\mathsf{kpca2},2\ell}, a\rangle|^2 \to 1$ and $|\langle \widehat{b}_{\mathsf{kpca2},2\ell}, b\rangle|^2 \to 1$, and thus $|\langle \widehat{u}_{\mathsf{kpca2},2\ell}, u\rangle|^2 \to 1$ which is the required result.

III.5.3 Stability of KPCA2,2r

The analysis of KPCA2,2r is very similar to that of KPCA2,2ℓ. The rearrangement in Step 1 yields

$$\mathcal{R}(X) = \sigma_0 \operatorname{vec}(as^H)\, b^T + \mathcal{R}(Z).$$

By Corollary III.1, and computations as in (3.31), substituting $\widehat{a}_{\mathsf{kpca2},2\ell} = \widehat{u}_1$, $a = \operatorname{vec}(as^H)$, $\widehat{v} = \widehat{b}_{\mathsf{kpca2},2\mathsf{r}}$ and $\operatorname{vec}(bs^H) = b$, we get for any $\delta_{2r,1} \in (0, \sigma_1(\mathcal{R}(X))]$:

$$\max\{1 - |\langle \widehat{u}_1, \operatorname{vec}(as^H)\rangle|^2, 1 - |\langle \widehat{b}_{\mathsf{kpca2},2\mathsf{r}}, b\rangle|^2\} \le 2\, \frac{||\mathcal{R}(Z)||_F^2}{\delta_{2r,1}^2}, \tag{3.37}$$

We highlight here that the permissible range of values for $\delta_{2r,1}$ is different from that for $\delta_{2\ell,1}$ in (3.31), since the dimensions of the rearranged data matrix are different.

Expressing $\widehat{u}_1$ in terms of $u_1 := \operatorname{vec}(as^H)$, and U_2, an orthonormal basis for the orthocomplement of u_1, as in (3.32) through (3.33), we get

$$\widehat{u}_1 = (u_1^H \widehat{u}_1)\, u_1 + U_2\, (U_2^H \widehat{u}_1).$$

Due to reshape($\widehat{u}_1$) in Step 4, we get:

$$\widehat{U}_1 := \operatorname{reshape}(\widehat{u}_1) = (u_1^H \widehat{u}_1)\, a\, s^H + \operatorname{reshape}\left(U_2\, U_2^H\, \widehat{u}_1\right).$$

Using similar computations as in (3.34) through (3.36), with $\widehat{b}_{\mathsf{kpca2},2\ell} = \widehat{a}_{\mathsf{kpca2},2\mathsf{r}}$, $b = a$, $v = u_1$, and $V_2 = U_2$, we obtain for all $\delta_{2r,1} \in (0, \sigma_1(\mathcal{R}(X))]$, $\delta_{2r,2} \in (0, \sigma_1(\mathcal{R}(\widehat{u}))]$:

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2},2\mathsf{r}}, a\rangle|^2, 1 - |\langle \widehat{s}, s\rangle|^2\} \le \frac{||Z||_F^2}{\delta_{2r,1}^2\, \delta_{2r,2}^2}. \tag{3.38}$$

From (3.37), (3.38), we see that as $||Z||_F \to 0$, $|\langle \widehat{b}_{\mathsf{kpca2},2\mathsf{r}}, b\rangle|^2 \to 1$ and $|\langle \widehat{a}_{\mathsf{kpca2},2\mathsf{r}}, a\rangle|^2 \to 1$ so that $|\langle \widehat{u}_{\mathsf{kpca2},2\mathsf{r}}, u\rangle|^2 \to 1$ as $||Z||_F \to 0$, which is the required result.

III.5.4 Stability of KPCA2,3

In KPCA2,3, we have that $\widehat{a}_{\mathsf{kpca2,3}} = \widehat{a}_{\mathsf{kpca2,2}\ell}$, and $\widehat{b}_{\mathsf{kpca2,3}} = \widehat{b}_{\mathsf{kpca2,2r}}$. The stability of KPCA2,3 thus follows from the stability of KPCA3,2ℓ and KPCA2,2r. Combining (3.31) and (3.37), we have that for all $\delta_{2\ell,1} \in (0, \sigma_1(\mathcal{R}_{m_1\, m_2 \times n \to m_1 \times m_2\, n}(X))]$ and $\delta_{2r,1} \in (0, \sigma_1(\mathcal{R}_{m_1\, m_2 \times n \to m_1\, n \times m_2}(X))]$:

$$\max\{1 - |\langle \widehat{a}_{\mathsf{kpca2,3}}, a\rangle|^2, 1 - |\langle \widehat{b}_{\mathsf{kpca2,3}}, b\rangle|^2\} \leq \left\{ \frac{||\mathcal{R}(Z)||_F^2}{\delta_{2\ell,1}^2}, \frac{||\mathcal{R}(Z)||_F^2}{\delta_{2r,1}^2} \right\}. \tag{3.39}$$

Since we have shown that as $||Z||_F \to 0$, $|\langle \widehat{a}_{\mathsf{kpca2,2}\ell}, a\rangle|^2 \to 1$ in Section III.5.2, and $|\langle \widehat{b}_{\mathsf{kpca2,2r}}, b\rangle|^2 \to 1$ in Section III.5.3, it follows that $|\langle \widehat{u}_{\mathsf{kpca2,3}}, u\rangle|^2 \to 1$ which completes the proof. □

III.6 Performance Analysis and Limits

In this section, we derive analytical expressions which quantify the estimation performance of the KPCA2 algorithms from Section III.4. To simplify exposition, we define the following functions:

Definition III.1. Let

$$\alpha^2(\sigma, c) := \begin{cases} 1 - \dfrac{c(1+\sigma^2)}{\sigma^2\,(\sigma^2 + c)} & \text{if } \sigma > c^{1/4} \\ 0 & \text{otherwise} \end{cases}, \tag{3.40a}$$

$$\beta^2(\sigma, c) := \begin{cases} 1 - \dfrac{c+\sigma^2}{\sigma^2\,(\sigma^2 + 1)} & \text{if } \sigma > c^{1/4} \\ 0 & \text{otherwise} \end{cases}. \tag{3.40b}$$

Proposition III.2. *Suppose that we observe Y as in (3.1). Let $\widehat{u}_{\mathsf{svd}}$ and $\widehat{v}_{\mathsf{svd}}$ be the principal left and right singular vectors of Y. Then, we have that as $m, n \to \infty$, $m/n \to c_{\mathsf{svd}} \in (0, \infty)$:*

$$|\langle \widehat{u}_{\mathsf{svd}}, u\rangle|^2 \xrightarrow{\text{a.s.}} \alpha^2(\sigma_0, c_{\mathsf{svd}}), \tag{3.41a}$$

$$|\langle \widehat{v}_{\mathsf{svd}}, v\rangle|^2 \xrightarrow{\text{a.s.}} \beta^2(\sigma_0, c_{\mathsf{svd}}). \tag{3.41b}$$

Proof. The proof is given in [14, Theorem 2.10]. □

Equation (3.41) tells us that the estimates $\widehat{u}_{\mathsf{svd}}$ and $\widehat{v}_{\mathsf{svd}}$ are correlated with the underlying vectors (i.e. are "informative") only if the SNR is above a "critical SNR" given by

$$\sigma_{\mathsf{crit,svd}} = c_{\mathsf{svd}}^{1/4}. \tag{3.42}$$

Below this critical SNR, the estimates $\widehat{u}_{\mathsf{svd}}$ will be uncorrelated with the underlying singular vector u, and thus uninformative. Denote

$$\alpha_{\mathsf{svd}}^2 := 1 - \frac{c_{\mathsf{svd}}(1+\sigma_0^2)}{\sigma_0^2(\sigma_0^2 + c_{\mathsf{svd}})} = \alpha^2(\sigma_0, c_{\mathsf{svd}}) \text{ when } \sigma_0 > c_{\mathsf{svd}}^{1/4}. \tag{3.43}$$

We will now provide quantification similar to (3.41) for the KPCA algorithms. To do so, we will assume that the following conjectures hold.

Conjecture III.1. *Suppose the hypothesis of Proposition III.2 hold. Then, we conjecture that,*

$$\sqrt{m}(\widehat{u}_{\mathsf{svd}} - \alpha_{\mathsf{svd}} u) \Rightarrow \mathcal{CN}\left(0, (1-\alpha_{\mathsf{svd}}^2)\, I_m\right) \tag{3.44}$$

where α_{svd} is given by (3.41), and $\Rightarrow$ denotes convergence in distribution.

Conjecture III.2. *Assume Conjecture III.1 holds. Then,*

$$\widehat{u}_{\mathsf{svd}} = \alpha_{\mathsf{svd}}\, u + \sqrt{\frac{1-\alpha_{\mathsf{svd}}^2}{m}}\, g + \Delta_u, \tag{3.45}$$

where Δ_u contains the higher order terms and $g \sim \mathcal{CN}(0, I)$. We conjecture that $\|\Delta_u\|_2 \xrightarrow{\text{a.s.}} 0$ as $m \to \infty$.

Remark. Assume that Conjectures III.1 and III.2 hold. Then, for large enough m, we may use the following approximation.

$$\widehat{u}_{\mathsf{svd}} \approx \alpha_{\mathsf{svd}}\, u + \sqrt{\frac{1-\alpha_{\mathsf{svd}}^2}{m}}\, g. \tag{3.46}$$

Claim III.1. *Assume that the hypotheses of Proposition III.2 hold, and that Conjectures III.1 and III.2 hold. Then, we have that, as $m_1, m_2, n \to \infty$, $m/n \to c_{\mathsf{svd}}$, $m_1/n \to c_p$, and $m_2/n \to c_q$,*

$$|\langle \widehat{u}_{\mathsf{kpca2,1}}, u\rangle|^2 \xrightarrow{\text{a.s.}} 1, \qquad \text{if } \sigma_0 > \sigma_{\mathsf{crit,kpca2,1}}, \tag{3.47a}$$

$$|\langle \widehat{u}_{\mathsf{kpca2,2}\ell}, u\rangle|^2 \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_p)}\, \alpha^2\left(\sqrt{\frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}}, c_q\right), \qquad \text{if } \sigma_0 > \sigma_{\mathsf{crit,kpca2,2}\ell}, \tag{3.47b}$$

$$|\langle \widehat{u}_{\mathsf{kpca2,2r}}, u\rangle|^2 \xrightarrow{\text{a.s.}} \alpha^2\left(\sqrt{\frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}}, c_p\right) \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_q)}, \qquad \text{if } \sigma_0 > \sigma_{\mathsf{crit,kpca2,2r}}, \tag{3.47c}$$

$$|\langle \widehat{u}_{\mathsf{kpca2,3}}, u\rangle|^2 \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_p)} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_q)}, \qquad \text{if } \sigma_0 > \sigma_{\mathsf{crit,kpca2,3}}, \tag{3.47d}$$

where

$$\begin{aligned}
\sigma_{\mathsf{crit,kpca2,1}} &= c_{\mathsf{svd}}^{1/4}, \\
\sigma_{\mathsf{crit,kpca2,2}\ell} &= \max\left\{c_{\mathsf{svd}}^{1/4}, \sqrt{\frac{\sqrt{c_q} + \sqrt{c_q + 4c_p\sqrt{c_q} + 4c_{\mathsf{svd}}}}{2}}\right\}, \\
\sigma_{\mathsf{crit,kpca2,2r}} &= \max\left\{c_{\mathsf{svd}}^{1/4}, \sqrt{\frac{\sqrt{c_p} + \sqrt{c_p + 4c_q\sqrt{c_p} + 4c_{\mathsf{svd}}}}{2}}\right\}, \\
\sigma_{\mathsf{crit,kpca2,3}} &= c_{\mathsf{svd}}^{1/4}.
\end{aligned}$$

Proof. See Appendix C. □

Discussion. In addition to c_{svd}, the expression for critical SNR of KPCA2,2ℓ and KPCA2,2r depend upon the aspect ratios m_1/n and m_2/n, which are the ratios of the dimensions of each Kronecker product component of u to the number of snapshots. Consider the case of KPCA2,2ℓ. There are three possible scenarios:

1. $m_1 = O(\sqrt{n})$, $m_2 = O(\sqrt{n})$, so that $c_p = 0$, $c_q = 0$. Substituting these limits into the expression for critical SNR, we see that $\sigma_{\mathsf{crit,kpca2,2}\ell} = c_{\mathsf{svd}}^{1/4}$, which is the same as that for SVD.

2. $m_1 = O(n)$, $m_2 = O(1)$, so that $c_p = O(1)$ and $c_q \to 0$. Substituting these limits into the expression for critical SNR, we once again obtain that $\sigma_{\mathsf{crit,kpca2}\ell} = c_{\mathsf{svd}}^{1/4}$.

3. $m_1 = O(1)$, $m_2 = O(n)$, so that $c_p \to 0$ and $c_q = O(1)$. Substituting these limits into (C.16), we obtain that $\sigma_{\mathsf{crit,bkpca2}\ell} = (\sqrt{c_q} + \sqrt{c_q + 4c_{\mathsf{svd}}})^{1/2}/2$, a value that is at least as large as $c_{\mathsf{svd}}^{1/4}$. This implies that the critical SNR of KPCA2,2ℓ is greater than that of SVD and KPCA2,1.

III.7 Numerical Simulations

In Fig. 3.1, we plot $|\langle \widehat{u}, u\rangle|^2$, the inner-product-squared estimation accuracy, for $\widehat{u} = \widehat{u}_{\mathsf{svd}}$, $\widehat{u}_{\mathsf{kpca2},1}$, $\widehat{u}_{\mathsf{kpca2},2\ell}$, $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$ and $\widehat{u}_{\mathsf{kpca2},3}$, for a system with $m_1 = 15$, $m_2 = 15$ and $n = 200$ in (3.1), as a function of SNR σ_0. The empirical plots were generated by generating data as given by (3.1), computing $\widehat{u}$ using KPCA2 algorithms, and then and computing the inner–product–squared over 10^4 trials. This plot shows that KPCA gives us better estimates of the underlying singular vector u compared to the SVD estimate, $\widehat{u}_{\mathsf{svd}}$. Furthermore, our theoretical predictions of estimation accuracy are in agreement with the empirically observed accuracies.

The theoretical plots were plotted as follows. The expressions for KPCA,1 are given by (C.3). For KPCA,2ℓ, (C.10a) was used as an approximation for $|\widehat{a}^H_{\mathsf{kpca2},2\ell} a|^2$ and and $|\widehat{a}^H_{\mathsf{kpca2},3} a|^2$, while (C.14) was used as an approximation for $|\widehat{b}^H_{\mathsf{kpca2},2\ell} b|^2$, with the value of $\sigma_{b,2\ell}$ in (C.14) set from (C.12). Similarly, for KPCA,2r, (C.21) was used as an approximation for $|\widehat{a}^H_{\mathsf{kpca2},2\mathsf{r}} a|^2$, with the value of $\sigma_{a,2r}$ in (C.18b) set from (C.19), and (C.18b) was used as an approximation for $|\widehat{b}^H_{\mathsf{kpca2},2\mathsf{r}} b|^2$ and $|\widehat{b}^H_{\mathsf{kpca2},3} b|^2$. In summary,

$$|\langle \widehat{u}_{\mathsf{kpca2},1}, u\rangle|^2 \approx \alpha^2\left(\sigma_{\mathrm{kpca},1},\, m_1/m_2\right)\, \beta^2\left(\sigma_{\mathrm{kpca},1},\, m_1/m_2\right), \tag{3.49a}$$

$$|\langle \widehat{u}_{\mathsf{kpca2},2\ell}, u\rangle|^2 \approx \alpha^2\left(\frac{\sigma_0}{\sqrt{m_2}}, \frac{m_1}{m_2 n}\right)\, \alpha^2\left(\sigma_{b,2\ell},\, \frac{m_2}{n}\right), \tag{3.49b}$$

$$|\langle \widehat{u}_{\mathsf{kpca2},2\mathsf{r}}, u\rangle|^2 \approx \beta^2\left(\sigma_0\sqrt{\frac{n}{m_2}}, \frac{m_1 n}{m_2}\right)\, \alpha^2\left(\sigma_{a,2r},\, \frac{m_2}{n}\right), \tag{3.49c}$$

$$|\langle \widehat{u}_{\mathsf{kpca2},3}, u\rangle|^2 \approx \alpha^2\left(\frac{\sigma_0}{\sqrt{m_2}}, \frac{m_1}{m_2 n}\right)\, \beta^2\left(\sigma_0\sqrt{\frac{n}{m_2}}, \frac{m_1 n}{m_2}\right). \tag{3.49d}$$

III.7.1 Application: STAP

The model (3.1) is motivated by applications such as Moving Target Indicator (MTI) radar, in which [16, 55]

$$u := u(\theta_c, f_c) = a(f_c) \otimes b(\theta_c) \tag{3.50}$$

where

$$a(f_c) = \frac{1}{\sqrt{m_1}}\begin{bmatrix} 1 & e^{j2\pi f_d} & \dots & e^{j2\pi(m_1-1)f_d}\end{bmatrix}^T, \tag{3.51a}$$

$$b(\theta_c) = \frac{1}{\sqrt{m_2}} \begin{bmatrix} 1 & e^{j2\pi\theta_c} & \dots & e^{j2\pi(m_2-1)\theta_c} \end{bmatrix}^T . \tag{3.51b}$$

In (3.51), f_d is the Doppler shift induced by clutter relative to the array boresight, and θ_c is the angular location of the clutter. An estimate of the vector u is explicitly utilized in the beamformer, which is given by [24, 16]

$$\widehat{w} = \frac{\widehat{\Sigma}^{-1} u(\theta_t, f_t)}{\|\widehat{\Sigma}^{-1} u(\theta_t, f_t)\|_2}, \tag{3.52}$$

where θ_t is the array look direction, and f_t the Doppler frequency induced at the target by the platform. In (3.52),the matrix $\widehat{\Sigma}$ is an estimate of the interference–plus–noise covariance, and is computed as

$$\widehat{\Sigma} = I + \sigma^2 \widehat{u}\,\widehat{u}^H, \tag{3.53}$$

and $\widehat{u}$ is an estimate of $u(\theta_c, f_c)$. The notch depth at angle θ and Doppler frequency f due to the filter $\widehat{w}$ is given by

$$\eta = \left|\widehat{w}^H u(\theta, f)\right|^2 . \tag{3.54}$$

We considered the Moving Target Indicator (MTI) application to illustrate the improvement in performance due to the KPCA estimates. The DMR beamformer [67], given by (3.52), is computed using the singular vector estimate, which is also the estimate of the STAP manifold vector (3.50). We simulated a system with $m_2 = 15$ sensors in a uniform linear array spaced half wavelength $(0.5 \times 0.67m)$ apart, which emits $m_1 = 15$ pulses in one coherent pulse interval. The target was located at $\theta_t = 0$, and induces a normalized Doppler shift $f_t = 0$. A single clutter was located at $\theta_c = 30^o$, and induced normalized Doppler shift $f_c = 0.0995$ at the array. In this application, σ_0 in (3.1) has the interpretation of clutter-to-noise ratio (CNR), and the goal of STAP is to suppress clutter.

We plot the notch depth of the MVDR filter, given by (3.54), as a function of clutter-to-noise ratio (CNR) in Fig. 3.2. Forming the MVDR filter using estimates from the KPCA2 algorithms result in deeper notch depths relative to forming the MVDR filter using the SVD estimate. The gain in the notch depth is plotted in Fig. 3.3. The deeper notch due to the KPCA estimates leads to improved clutter suppression compared to the SVD estimates - e.g. the notch depth at the interferer location using KPCA2 estimates is 10 dB deeper than the notch depth using SVD at SNR = 20 dB. Predictably, KPCA2,1 gives the best notch depth gain.

In Fig. 3.4, we fix the target at $\theta_t = 0^o$, and and vary the location of the single clutter, θ_c.

The plot compares the the notch depth at clutter location for the four KPCA2 algorithms. A deeper notch depth indicates better clutter suppression. We see that KPCA2,1 gives us the best clutter suppression uniformly over the range of interferer locations, and KPCA2,3 the worst. The notch depths due to KPCA2,2ℓ and KPCA2,2r lie in between these two values, but neither algorithm is uniformly better than the other.

In Fig. 3.5, we plot the inner-product-squared accuracy of the singular vector estimate $\widehat{u}_{\mathsf{kpca2},2\ell}$ relative to the true vector u, over a range of SNRs and "aspect ratios" m/n, which is the ratio of dimensions of u to the number of snapshots. We see a sharp phase transition in algorithm performance, around the value predicted by (C.10). Though the analysis leading to (C.10) is asymptotic in nature, we see that the prediction is in good agreement with empirical values for reasonably sized arrays.

In Fig. 3.6, we plot the notch depth gain against clutter-to-noise ratio and clutter location. We see using KPCA2,1 gives us a gain in notch depth over using the SVD in the entire range of clutter locations and CNRs, above the critical value. In Fig. 3.7, we compare the estimation performance of KPCA2,1 against the LowRank-Kronecker PCA (LR-Kron) algorithm introduced by Hero and Greenwald in [2]. We see that KPCA2,1 and LR-Kron give similar estimation performance. In contrast to LR-Kron, which is an iterative algorithm which requires an eigendecomposition in each iteration, KPCA2,1 requires exactly two SVD operations.

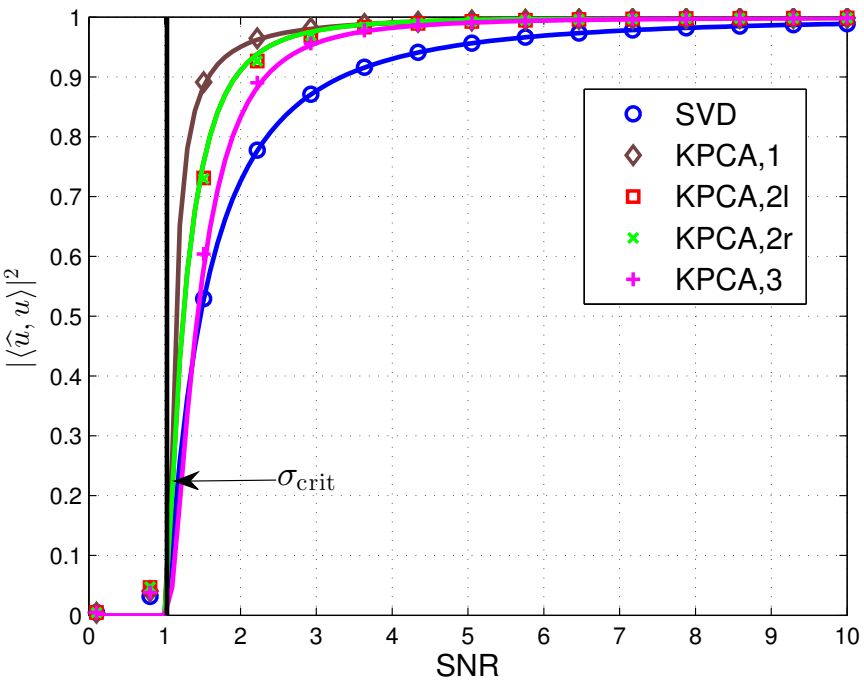


Figure 3.1: Plot of inner-product-squared accuracy, $|\langle \widehat{u}, u \rangle|^2$ vs SNR (σ), where $\widehat{u} = \widehat{u}_{\mathsf{svd}}$, $\widehat{u}_{\mathsf{kpca2},2\ell}$, $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$, $\widehat{u}_{\mathsf{kpca2},3}$, for $m_1 = 15$, $m_2 = 15$ and $n = 200$ in (3.1). The discrete points are obtained by averaging empirical values over 10^4 trials. The continuous plots are obtained via the theoretical expressions in (3.47). For a given SNR, $\widehat{u}_{\mathsf{kpca2},1}$ has the highest accuracy, greater than $\widehat{u}_{\mathsf{kpca2},2\ell}$ and $\widehat{u}_{\mathsf{kpca2},2\mathsf{r}}$ which have equal accuracy, greater than $\widehat{u}_{\mathsf{kpca2},3}$. All KPCA2 algorithms have higher accuracy than the SVD estimate. The theoretical predictions are in good agreement with the empirical values. $\sigma_{\text{crit}} = c_{\text{svd}}^{1/4}$ is the phase transition boundary given by (3.41).

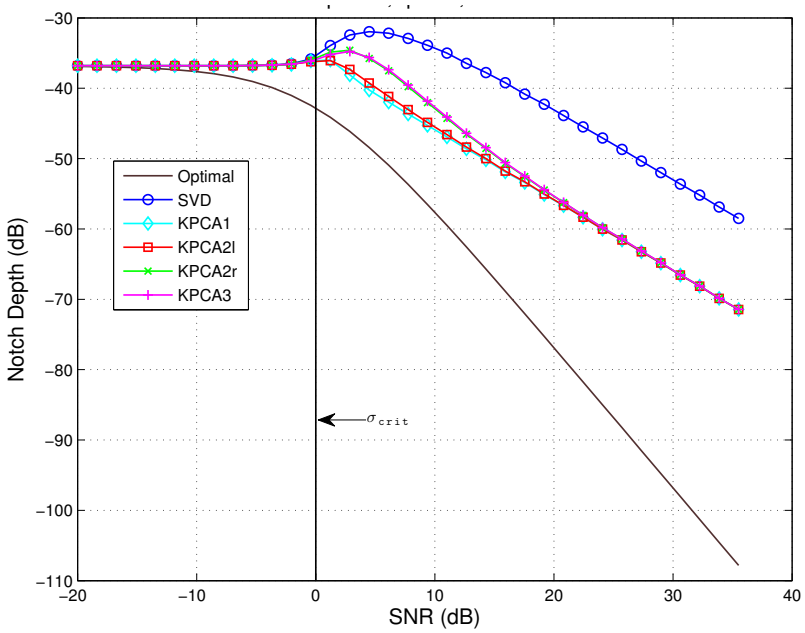


Figure 3.2: Application to STAP: Plot of Notch Depth (dB) given by (3.54) vs SNR (dB), where $\widehat{\Sigma}$ in (3.53) (and thus $\widehat{w}$ in (3.52)) is estimated using $u(\theta_c, f_c)$ (Optimal), and $\widehat{u} = \widehat{u}_{\mathsf{svd}}$, $\widehat{u}_{\mathsf{kpca2,1}}$, $\widehat{u}_{\mathsf{kpca2,2}\ell}$, $\widehat{u}_{\mathsf{kpca2,2r}}$, $\widehat{u}_{\mathsf{kpca2,3}}\}$. The array look angle is 0^o and the interferer is at 30^o. The dimensions of the system are as in Fig. 3.1. All four versions of the KPCA2 algorithm suppress the interference better than the SVD, e.g. at SNR = 20dB, we obtain 10dB better suppression using KPCA2. KPCA1 has the deepest notch depth, followed by KPCA2,2ℓ, KPCA2,2r and then KPCA2,3.

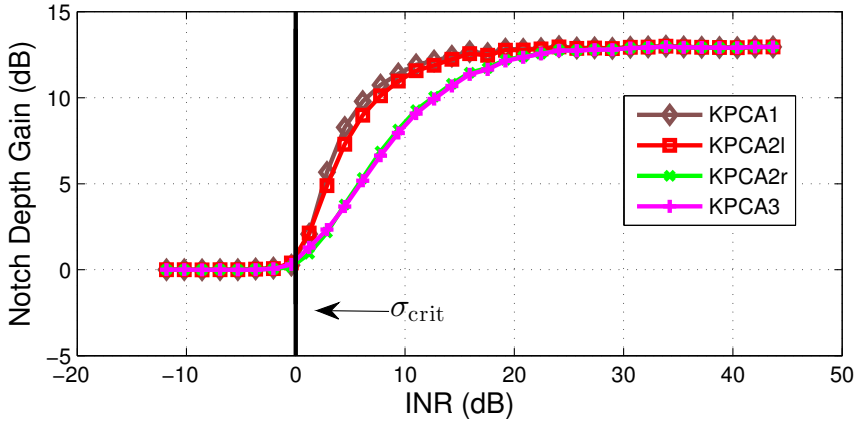


Figure 3.3: Application to STAP: Notch depth gain of the KPCA2 algorithms relative to the SVD algorithms vs INR, for the system in Fig. 3.1. We see that all KPCA2 algorithms give us notch depth gain relative to the SVD. KPCA2,1 performs the best, and KPCA2,3 gives the least notch depth gain. As expected from Fig. 3.4, KPCA2,2l and KPCA2,2r give gains which are in between the other two algorithms. All KPCA2 algorithms give 13 dB gain in the high INR regime.

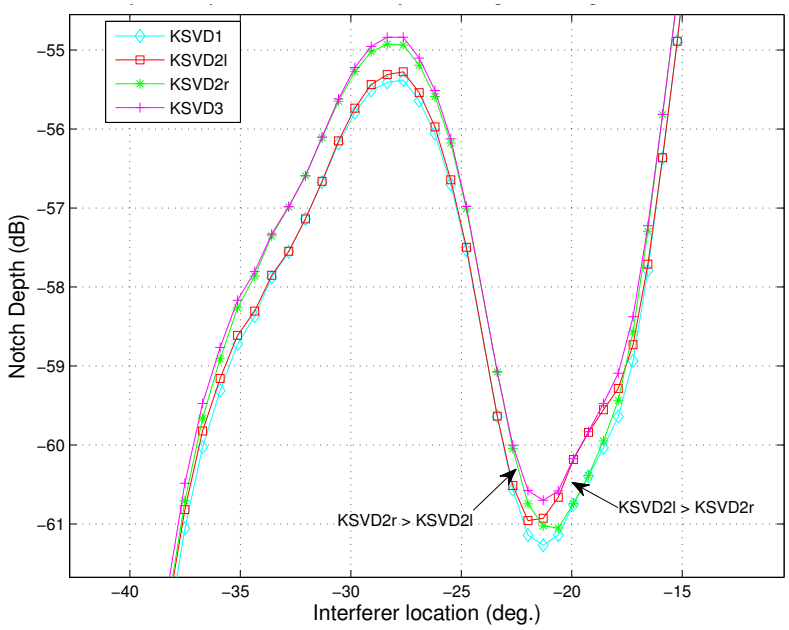


Figure 3.4: Application to STAP: Notch Depth (dB) of the four KPCA2 algorithms vs interferer location (deg). This plot highlights the relative performance of the KPCA2 algorithms, which may be observed in Fig. 3.1 and 3.2. The dimensions of the system are as in Fig. 3.1. KPCA2,1 has the deepest notch depth at all locations of the interferer, and KPCA2,3 has the shallowest notch depth. The notch depths due to KPCA2,2l and KPCA2,2r are in between those of the other two variants, but either one does not suppress interference uniformly better than the other. For example, KPCA2,2ℓ has a deeper notch depth at -23^o (relative to the array broadside), while KPCA2,2r has a deeper notch depth at -18^o.

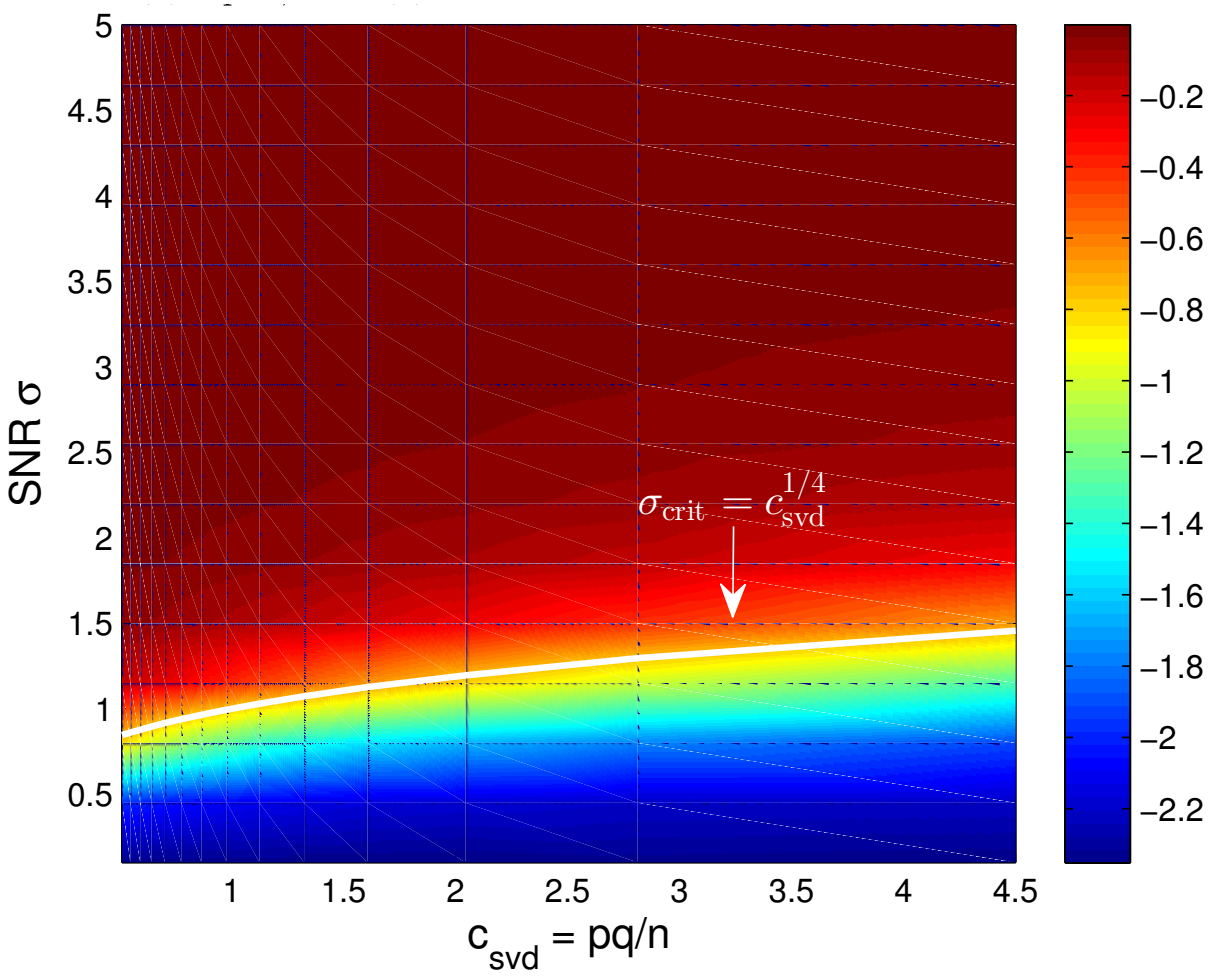


Figure 3.5: Heatmap of the logarithm of empirical inner-product-squared accuracy versus aspect ratio $c_{\text{svd}} = m/n$ and SNR σ, for the system in Fig. 3.1. In the 'hot' region (red), $\widehat{u}_{\text{kpca2},2\ell}$ is highly correlated with u, and in the cold region (blue), estimation performance of $\widehat{u}_{\text{kpca2},2\ell}$ is equivalent to choosing a random unit vector on the n dimensional sphere. We see a sharp phase transition boundary between the two regimes, and the theoretical boundary prediction given by (3.47) and (3.41) and accurately predicts the empirically observed boundary.

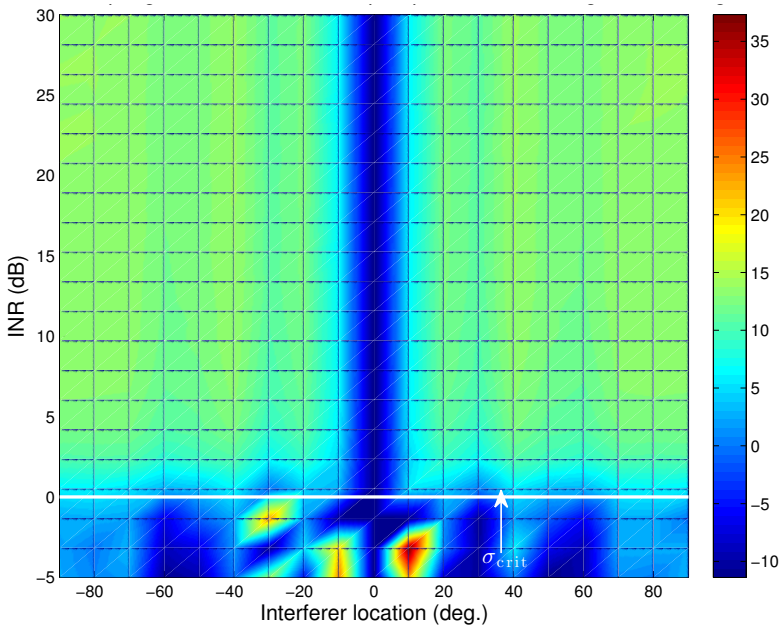


Figure 3.6: Application to STAP: Heatmap of the notch depth gain due to KPCA2,1 relative to the SVD, for the same setup as in Fig. 3.1, as a function of interferer location and INR. The target is located at 0^o. We obtain around 15 db gain throughout the region away from the target location.

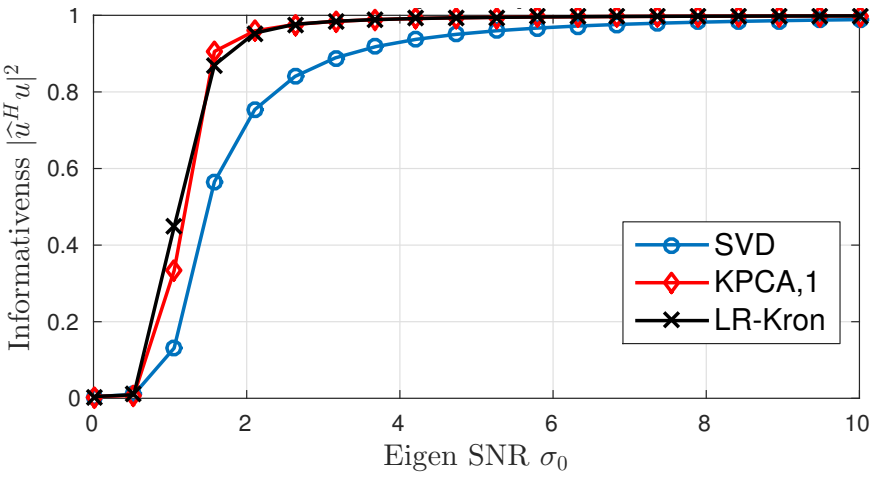


Figure 3.7: Plot of empirical estimation performance of KPCA2,1 versus the LowRank-KroneckerPCA (LR-Kron) algorithm introduced in [2], for the same setup as in Fig. 3.1. The plot corresponding to "SVD" is the informativeness of $\widehat{u}_{\text{svd}}$. We see KPCA2,1 and LR-Kron yield identical estimation performance. However, KPCA2,1 prescribes a more computationally direct algorithm involving exactly two SVDs and one rearrangement operation to find the singular vector estimate, whereas LR-Kron is an iterative algorithm which requires two eigendecompositions in each iteration of the algorithm, and the total number of iterations depends upon the rate at which the residual error decreases in an alternate minimization setup.

III.8 Conclusions

In this work, we introduced four new algorithms to estimate the Kronecker structured singular vectors of low rank signals buried in noise. The algorithms exploit the Kronecker structure of the underlying singular vectors, and yield more accurate estimates compared to the canonical estimate computed using the SVD of the data. We showed that the improved estimates better suppress clutter in the STAP application. Using matrix perturbation theory, we showed that all of the newly proposed algorithms are stable. We found that first computing the singular vector of the data, and then cleaning up the estimate to fit the Kronecker structure yields better estimates than first rearranging the data to exploit the Kronecker structure which leads to a performance loss. Using recent results from random matrix theory, we provided justification for this relative order of algorithm performance. The rigorous characterization of the fluctuations of singular vectors of random matrices such as those in Conjecture III.1 remain important open problems.

CHAPTER IV

Algorithms for Estimation of Rank One Signal in Noise with Triple Kronecker Product Structured Singular Vectors

IV.1 Introduction

Signals which have Kronecker structured singular vectors arise in many important applications in signal processing. Here we consider low rank signals whose left singular vectors are Kronecker products of some unknown unit norm vectors. Applications of this signal model include Space-Time Adaptive Processing (STAP), Multiple-Input Multiple-Output (MIMO) STAP, Vector Sensor Array applications, etc. Estimation of the singular vectors of such signals buried in high dimensional noise is canonically done using the Singular Value Decomposition (SVD) of the data. However, the SVD does not take into account the Kronecker structure of the singular vectors. One might expect that algorithms which explicitly account for the Kronecker structure of the singular vectors will lead to improved estimation performance.

In recent work, Tsiligkaridis and Hero [56] considered the estimation of full-rank covariance matrices which are Kronecker products of full-rank matrices. They developed an algorithm which uses a rearrangement operator, developed by Van Loan and Pistianis [19], that enables decomposition of a matrix into a sum of Kronecker product components. In their algorithm, they first rearrange the data using the rearrangement operator, and then employ SVD based estimation techniques. In [68], the authors consider a slightly different model, in which only the left singular vector of the signal has a Kronecker product form. They propose estimation algorithms which first use the SVD to estimate the singular vectors,

and then fit a Kronecker structure to the SVD based estimates, as well as algorithms which first apply the rearrangement operator on the data and then compute SVD based estimates.

This work is an extension of [68] to signals with triple Kronecker structured left singular vectors. We propose eight new algorithms, called KPCA3,1 – KPCA3,8 (collectively referred to as "KPCA3") to estimate the Kronecker product structured singular vector. All algorithms utilize the rearrangement operator in different ways. These algorithms are finite (non-iterative), and they each involve a three stage optimization problem. The solution to every stage of the optimization gives us an estimate of a Kronecker product component of the singular vector, and involves an SVD computation. Hence all our algorithms involve three SVD computations. We study the relative estimation performance of the algorithms based on recent results from random matrix theory, and we demonstrate improved estimation performance in the MIMO STAP application.

The rest of this chapter is organized as follows. In Section IV.2, we describe the model for our system, and highlight the connections between our model and a tensor signal model. In Section IV.3, we propose eight new algorithms which use the rearrangement operator to exploit the redundancy in the data due to the underlying Kronecker structure. In Section IV.4, we provide analytical expressions which quantify algorithms estimation performance. In Section IV.5, we present some numerical simulations and concluding remarks are in Section IV.6.

IV.2 Model and Estimation Problem

Consider the signal-plus-noise model

$$Y = \sigma \underbrace{(a \otimes b \otimes c)}_{:=u} s^H + Z \tag{4.1}$$

where observed data matrix $Y \in \mathbb{C}^{m \times n}$ contains n snapshots of m dimensional data. Here, $\sigma \in \mathbb{R}_+$ is the eigen-SNR; $s \in \mathbb{C}^n$; $u \in \mathbb{C}^{m:=m_1 m_2 m_3}$ is the Kronecker product ($\otimes$) of $a \in \mathbb{C}^{m_1}$, $b \in \mathbb{C}^{m_2}$, and $c \in \mathbb{C}^{m_3}$; and $Z \in \mathbb{C}^{m \times n}$ is the noise matrix, whose elements have mean 0, variance $1/m$, and bounded higher order moments.

Our goal is to estimate u from Y. The canonical estimate is found from the SVD of Y: if

$$Y = \widehat{\sigma}_{\mathsf{svd}} \widehat{u}_{\mathsf{svd}} \widehat{v}_{\mathsf{svd}}^H$$

is the (ordered) SVD of Y truncated to rank one, then $\widehat{u}_{\mathsf{svd}}$ is the canonical estimate of u. However, $\widehat{u}_{\mathsf{svd}}$ is oblivious to the Kronecker product structure of u. We present new algorithms for estimation of u which account for this structure, and therefore, are more accurate estimates of u.

IV.2.1 Connections to Low Rank Tensor Decomposition

The problem of estimating u is intimately linked to the the problem of low rank tensor estimation. Here we note only the details relevant to our problem.

Let

$$\mathcal{Y} = \sigma\, \bar{s} \circ c \circ b \circ a + \mathcal{Z}$$

where $\circ$ denotes outer-product, and $\mathcal{Y}, \mathcal{Z} \in \mathbb{C}^{n \times m_1 \times m_2 \times m_3}$ are tensors of order four. Let $\mathcal{Y}_{[j]}$ denote *unfolding* $\mathcal{Y}$ along mode j. The unfolding operation extracts fibers from mode j of $\mathcal{Y}$, and arranges them as columns of a matrix. Unfolding $\mathcal{Y}$ along mode 1,

$$\mathcal{Y}_{[1]} = \sigma\, \bar{s}\, (a \otimes b \otimes c)^T + \mathcal{Z}_{[1]}. \tag{4.2}$$

Comparing (4.2) with (4.1), we see that $\mathcal{Y}_{[1]} = Y^T$. Low rank estimation of Y is thus equivalent to the low rank estimation of $\mathcal{Y}$.

IV.3 New Algorithms for the Estimation of u

In this section, we introduce eight new algorithms for the estimation of u in (4.1), called KPCA3,1 (Algorithm 4.1) – KPCA3,8 (Algorithm 4.8). Each algorithm uses the rearrangement operator introduced in Section III.3 in a different way, and involves three SVD computations. As an illustrative example, we show the detailed derivation of KPCA3,4 (Algorithm

4.4); the other algorithms were derived analogously.

Algorithm 4.1: KPCA3,1

1 $\widehat{u}_{\mathsf{svd}}$ = principal left singular vector of Y /* 1st SVD */

2 $\widehat{U}_{\mathcal{R}}$ = $\mathcal{R}_{m_1\ m_2\ m_3 \times 1 \to m_1 \times m_2\ m_3}(\widehat{u}_{\mathsf{svd}})$

3 $\widehat{a}_{\mathsf{kpca3,1}}$ = principal left singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

4 $\widehat{v}^*$ = principal right singular vector of $\widehat{U}_{\mathcal{R}}$

5 $\widehat{V}_{\mathcal{R}}$ = $\mathcal{R}_{m_2\ m_3 \times 1 \to m_2 \times m_3}(\widehat{v})$

6 $\widehat{b}_{\mathsf{kpca3,1}}$ = principal left singular vector of $\widehat{V}_{\mathcal{R}}$ /* 3rd SVD */

7 $\widehat{c}^*_{\mathsf{kpca3,1}}$ = principal right singular vector of $\widehat{V}_{\mathcal{R}}$

Algorithm 4.2: KPCA3,2

1 $\widehat{u}_{\mathsf{svd}}$ = principal left singular vector of Y /* 1st SVD */

2 $\widehat{U}_{\mathcal{R}}$ = $\mathcal{R}_{m_1\ m_2\ m_3 \times 1 \to m_1\ m_2 \times m_3}(\widehat{u}_{\mathsf{svd}})$

3 $\widehat{c}^*_{\mathsf{kpca3,2}}$ = principal right singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

4 $\widehat{u}$ = principal left singular vector of $\widehat{U}_{\mathcal{R}}$

5 $\widehat{U}_{\mathcal{R},2}$ = $\mathcal{R}_{m_1\ m_2 \times 1 \to m_1 \times m_3}(\widehat{u})$

6 $\widehat{a}_{\mathsf{kpca3,2}}$ = principal left singular vector of $\widehat{U}_{\mathcal{R},2}$ /* 3rd SVD */

7 $\widehat{b}^*_{\mathsf{kpca3,2}}$ = principal right singular vector of $\widehat{U}_{\mathcal{R},2}$

Algorithm 4.3: KPCA3,3

$Y_{\mathcal{R}} = \mathcal{R}_{m_1\, m_2\, m_3 \times n \to m_1\, m_2 \times m_3\, n}(Y)$

$\widehat{u}$ = principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

$\widehat{v}^*$ = principal right singular vector of $Y_{\mathcal{R}}$

$\widehat{U}_{\mathcal{R}} = \mathcal{R}_{m_1\, m_2 \times 1 \to m_1 \times m_2}(\widehat{u})$

$\widehat{a}_{\mathsf{kpca3,3}}$ = principal left singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

$\widehat{b}^*_{\mathsf{kpca3,3}}$ = principal right singular vector of $\widehat{U}_{\mathcal{R}}$

$\widehat{V} = \text{reshape}_{m_3\, n \times 1 \to m_3 \times n}(\widehat{v})$

$\widehat{c}_{\mathsf{kpca3,3}}$ = principal left singular vector of $\widehat{V}$ /* 3rd SVD */

Algorithm 4.4: KPCA3,4

$Y_{\mathcal{R}} = \mathcal{R}_{m_1\, m_2\, m_3 \times n \to m_1 \times m_2\, m_3\, n}(Y)$

$\widehat{a}_{\mathsf{kpca3,4}}$ = principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

$\widehat{v}^*$ = principal right singular vector of $Y_{\mathcal{R}}$

$\widehat{V} = \text{reshape}_{m_2\, m_3\, n \times 1 \to m_2\, m_3 \times n}(\widehat{v})$

$\widehat{V}_{\mathcal{R}} = \mathcal{R}_{m_2\, m_3 \times n \to m_2 \times m_3\, n}(\widehat{V})$

$\widehat{b}_{\mathsf{kpca3,4}}$ = principal left singular vector of $\widehat{V}_{\mathcal{R}}$ /* 2nd SVD */

$\widehat{v}^*_2$ = principal right singular vector of $\widehat{V}_{\mathcal{R}}$

$\widehat{V}_2 = \text{reshape}_{m_3\, n \times 1 \to m_3 \times n}(\widehat{v}_2)$

$\widehat{c}_{\mathsf{kpca3,4}}$ = principal left singular vector of $\widehat{V}_2$ /* 3rd SVD */

Algorithm 4.5: KPCA3,5

$Y_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2\ m_3 \times n \to m_1\ m_2\ n \times m_3}(Y)$

$\widehat{c}^{*}_{\mathsf{kpca3,5}}$ = principal right singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

$\widehat{u}$ = principal left singular vector of $Y_{\mathcal{R}}$

$\widehat{U} = \text{reshape}_{m_1\ m_2\ n \times 1 \to m_1\ m_3 \times n}(\widehat{v})$

$\widehat{U}_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2 \times n \to m_1 \times m_2\ n}(\widehat{U})$

$\widehat{a}_{\mathsf{kpca3,5}}$ = principal left singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

$\widehat{v}^{*}$ = principal right singular vector of $\widehat{U}_{\mathcal{R}}$

$\widehat{V} = \text{reshape}_{m_2\ n \times 1 \to m_2 \times n}(\widehat{v})$

$\widehat{b}_{\mathsf{kpca3,5}}$ = principal left singular vector of $\widehat{V}$ /* 3rd SVD */

Algorithm 4.6: KPCA3,6

$Y_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2\ m_3 \times n \to m_1 \times m_2\ m_3\ n}(Y)$

$\widehat{a}_{\mathsf{kpca3,6}}$ = principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

$\widehat{v}^{*}$ = principal right singular vector of $Y_{\mathcal{R}}$

$\widehat{V} = \text{reshape}_{m_2\ m_3\ n \times 1 \to m_2\ m_3 \times n}(\widehat{v})$

$\widehat{V}_{\mathcal{R}} = \mathcal{R}_{m_2\ m_3 \times n \to m_2\ n \times m_3}(\widehat{V})$

$\widehat{c}^{*}_{\mathsf{kpca3,6}}$ = principal right singular vector of $\widehat{V}_{\mathcal{R}}$ /* 2nd SVD */

$\widehat{u}$ = principal right singular vector of $\widehat{V}_{\mathcal{R}}$

$\widehat{U} = \text{reshape}_{m_2\ n \times 1 \to m_2 \times n}(\widehat{u})$

$\widehat{b}_{\mathsf{kpca3,6}}$ = principal left singular vector of $\widehat{U}$ /* 3rd SVD */

Algorithm 4.7: KPCA3,7

1 $Y_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2\ m_3 \times n \to m_1\ n \times m_2\ m_3}(Y)$

2 $\widehat{u} =$ principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

3 $\widehat{v}^* =$ principal right singular vector of $Y_{\mathcal{R}}$

4 $\widehat{U} = \text{reshape}_{m_1\ n \times 1 \to m_1 \times n}(\widehat{u})$

5 $\widehat{a}_{\text{kpca3,7}} =$ principal left singular vector of $\widehat{U}$ /* 2nd SVD */

6 $\widehat{V} = \text{reshape}_{m_2\ m_3 \times 1 \to m_2 \times m_3}(\widehat{v})$

7 $\widehat{b}_{\text{kpca3,7}} =$ principal left singular vector of $\widehat{V}$ /* 3rd SVD */

8 $\widehat{c}^*_{\text{kpca3,7}} =$ principal right singular vector of $\widehat{V}$

Algorithm 4.8: KPCA3,8

1 $Y_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2\ m_3 \times n \to m_1\ m_2\ n \times m_3}(Y)$

2 $\widehat{u} =$ principal left singular vector of $Y_{\mathcal{R}}$ /* 1st SVD */

3 $\widehat{c}^*_{\text{kpca3,8}} =$ principal right singular vector of $Y_{\mathcal{R}}$

4 $\widehat{U} = \text{reshape}_{m_1\ m_2\ n \times 1 \to m_1\ m_2 \times n}(\widehat{u})$

5 $\widehat{U}_{\mathcal{R}} = \mathcal{R}_{m_1\ m_2 \times n \to m_1\ n \times m_2}(\widehat{U})$

6 $\widehat{b}^*_{\text{kpca3,8}} =$ principal right singular vector of $\widehat{U}_{\mathcal{R}}$ /* 2nd SVD */

7 $\widehat{u}_2 =$ principal left singular vector of $\widehat{U}_{\mathcal{R}}$

8 $\widehat{U}_2 = \text{reshape}_{m_1\ n \times 1 \to m_1 \times n}(\widehat{u}_2)$

9 $\widehat{a}_{\text{kpca3,8}} =$ principal left singular vector of $\widehat{U}_2$ /* 3rd SVD */

IV.3.1 Detailed Derivation of KPCA3,4

Recall from (4.1) that

$$\begin{aligned} Y &= \sigma\,(a \otimes b \otimes c)\, s^H + Z \\ &= \sigma\ (a \otimes (b \otimes c))\ (1 \otimes s^H) + Z \\ &= \sigma\ a \otimes (b \otimes c) s^H + Z. \end{aligned}$$

Our estimation problem is

$$\widehat{a}, \widehat{b}, \widehat{c}, \widehat{\sigma} = \underset{\substack{a,b,c,\sigma: \\ \|a\|_2=1,\, \|b\|_2=1,\, \|c\|_2=1,\, \sigma\in\mathbb{R}_+}}{\arg\min} \|Y - \sigma\ a \otimes (b \otimes c) s^H\|_F^2 \tag{4.3a}$$

$$= \underset{\substack{a,b,c,\sigma: \\ \|a\|_2=1,\, \|b\|_2=1,\, \|c\|_2=1,\, \sigma\in\mathbb{R}_+}}{\arg\min} \|Y_{\mathcal{R}} - \sigma\ \mathrm{vec}\,(a)\ \mathrm{vec}\left((b \otimes c) s^H\right)^T\|_F^2 \tag{4.3b}$$

where $Y_{\mathcal{R}} := \mathcal{R}_{m_1\ m_2\ m_3 \times n \to m_1 \times m_2\ m_3\ n}(Y)$ as defined in (3.4),

$$= \underset{\substack{a,b,c,\sigma: \\ \|a\|_2=1,\, \|b\|_2=1,\, \|c\|_2=1,\, \sigma\in\mathbb{R}_+}}{\arg\min} \|Y_{\mathcal{R}} - \sigma\ a\ \mathrm{vec}\left((b \otimes c) s^H\right)^T\|_F^2. \tag{4.3c}$$

Problem (4.3c) involves joint optimization over a, b, c, σ. In its stead, we propose solving a multi-stage optimization problem to obtain estimates of a, b, c.

Inspired by (4.3c), the first stage optimization problem is

$$\widehat{a}_{\mathsf{kpca3,4}}, \widehat{v} = \underset{\substack{a,v \\ \|a\|_2=1,\, \|v\|_2=1}}{\arg\min} \|Y_{\mathcal{R}} - a v^T\|_F^2.$$

and its solution is:

$$\begin{aligned} \widehat{a}_{\mathsf{kpca3,4}} &= \text{principal left singular vector of } Y_{\mathcal{R}}, \\ \widehat{v}^* &= \text{principal right singular vector of } Y_{\mathcal{R}}. \end{aligned} \tag{4.4}$$

Hence we obtain $\widehat{a}_{\mathsf{kpca3,4}}$, the estimate of a using KPCA3,4. Note that $\widehat{v}$ is an estimate of $\mathrm{vec}\left((b \otimes c) s^H\right)$ from (4.3c), and let

$$\delta_v := \widehat{v} - \mathrm{vec}\left((b \otimes c) s^H\right)$$

be the estimation error. Reshaping $\widehat{v}$ into a matrix, we get

$$\begin{aligned}\underbrace{\text{reshape}_{m_2\ m_3\ n\times 1\to m_2\ m_3\times n}(\widehat{v})}_{:=\widehat{V}} &= (b\otimes c)s^H - \underbrace{\text{reshape}_{m_2\ m_3\ n\times 1\to m_2\ m_3\times n}(\delta_v)}_{:=\Delta_v}\\ &= b\otimes cs^H - \Delta_v.\end{aligned}$$

Using the rearrangement operator from (3.4), we get

$$\begin{aligned}\widehat{V}_{\mathcal{R}} &:= \mathcal{R}_{m_2\ m_3\times n\to m_2\times m_3\ n}(\widehat{V})\\ &= b\ \text{vec}\left(cs^H\right) - \mathcal{R}_{m_2\ m_3\times n\to m_2\times m_3\ n}(\Delta_v).\end{aligned} \tag{4.5}$$

Inspired by (4.5), the second stage optimization problem is

$$\widehat{b}_{\mathsf{kpca3,4}}, \widehat{v}_2 = \underset{\substack{b,v\\ \|b\|_2=1,\ \|v\|_2=1}}{\arg\min} \|\widehat{V} - bv^T\|_F^2$$

and its solution is:

$$\begin{aligned}\widehat{b}_{\mathsf{kpca3,4}} &= \text{principal left singular vector of } \widehat{V},\\ \widehat{v}_2^* &= \text{principal right singular vector of } \widehat{V}.\end{aligned} \tag{4.6}$$

Hence we obtain $\widehat{b}_{\mathsf{kpca3,4}}$, the estimate of b using KPCA3,4. Note that $\widehat{v}_2$ is an estimate of $\text{vec}\left(cs^H\right)$ from (4.5), and let

$$\delta_{v2} := \widehat{v}_2 - \text{vec}\left(cs^H\right)$$

be the estimation error. Reshaping $\widehat{v}_2$ into a matrix, we get

$$\begin{aligned}\underbrace{\text{reshape}_{m_3\ n\times 1\to m_3\times n}(\widehat{v}_2)}_{:=\widehat{V}_2} &= cs^H - \underbrace{\text{reshape}_{m_3\ n\times 1\to m_3\times n}(\delta_{v2})}_{:=\Delta_{v2}}\\ &= cs^H - \Delta_{v2}.\end{aligned} \tag{4.7}$$

Inspired by (4.7), the third stage optimization problem is

$$\widehat{c}_{\mathsf{kpca3,4}} = \underset{\substack{c,v:\\ \|c\|_2=1,\ \|v\|_2=1}}{\arg\min} \|\widehat{V}_2 - cv^T\|_F^2$$

and its solution is:

$$\widehat{c}_{\mathsf{kpca3,4}} = \text{principal left singular vector of } \widehat{V}_2 \tag{4.8}$$

which is the estimate of c using KPCA3,4. Given estimates of the Kronecker product components of u in (4.4), (4.6) and (4.8), we form the following new estimate of u:

$$\widehat{u}_{\mathsf{kpca3,4}} = \widehat{a}_{\mathsf{kpca3,4}} \otimes \widehat{b}_{\mathsf{kpca3,4}} \otimes \widehat{c}_{\mathsf{kpca3,4}}. \tag{4.9}$$

IV.4 Performance Analysis and Limits

In this Section, we analyze the performance of the KPCA3 algorithms proposed in Section IV.3. To simplify exposition, we define the following functions.

Definition IV.1. Let

$$\alpha^2(\sigma, c) := 1 - \frac{c(1+\sigma^2)}{\sigma^2\,(\sigma^2+c)}, \tag{4.10a}$$

$$\beta^2(\sigma, c) := 1 - \frac{c+\sigma^2}{\sigma^2\,(\sigma^2+1)}. \tag{4.10b}$$

Proposition IV.1. *Suppose that we have Y as in (4.1). Let $\widehat{u}_{\mathsf{svd}}$ and $\widehat{v}_{\mathsf{svd}}$ be principal left and right singular vectors of X. Then, we have that,*

$$|\langle \widehat{u}_{\mathsf{svd}}, u\rangle|^2 \xrightarrow{\text{a.s.}} \begin{cases} \alpha^2(\sigma_0, c_0) & \text{if } \sigma_0 > c_0^{1/4} \\ 0 & \text{else} \end{cases}. \tag{4.11a}$$

$$|\langle \widehat{v}_{\mathsf{svd}}, v\rangle|^2 \xrightarrow{\text{a.s.}} \begin{cases} \beta^2(\sigma_0, c_0) & \text{if } \sigma_0 > c_0^{1/4} \\ 0 & \text{else} \end{cases}. \tag{4.11b}$$

as $m, n \to \infty$, $m/n \to c_0$.

Proof. The proof is given in [14, Theorem 2.10]. □

Equation (4.11) tells us that the estimates $\widehat{u}_{\mathsf{svd}}$ and $\widehat{v}_{\mathsf{svd}}$ are correlated with the underlying vectors (i.e. are "informative") only if the SNR is above a "critical SNR" given by

$$\sigma_{\mathsf{crit,svd}} = c_0^{1/4}. \tag{4.12}$$

Below this critical SNR, the estimates will be uncorrelated with the corresponding underlying

singular vector, and thus "uninformative". We will now provide quantification similar to (4.11) for the KPCA algorithms, assuming the following conjectures hold.

Conjecture IV.1. *Suppose the hypothesis of Proposition IV.1 hold. Then, we conjecture that,*

$$\sqrt{m}(\widehat{u}_{\text{svd}} - \alpha_{\text{svd}} u) \Rightarrow \mathcal{CN}\left(0, (1 - \alpha_{\text{svd}}^2)\, I_m\right), \tag{4.13a}$$

$$\sqrt{m}(\widehat{u}_{\text{svd}} - \alpha_{\text{svd}} u) \Rightarrow \mathcal{CN}\left(0, (1 - \alpha_{\text{svd}}^2)\, I_m\right), \tag{4.13b}$$

where

$$\alpha_{\text{svd}}^2 := \alpha^2(\sigma_0, c_0), \tag{4.14a}$$

$$\beta_{\text{svd}}^2 := \beta^2(\sigma_0, c_0), \tag{4.14b}$$

and $\Rightarrow$ *denotes convergence in distribution.*

Conjecture IV.2. *Assume Conjecture IV.1 holds. Then,*

$$\widehat{u}_{\text{svd}} = \alpha_{\text{svd}}\, u + \sqrt{\frac{1 - \alpha_{\text{svd}}^2}{m}}\, g + \Delta_u,$$

$$\widehat{v}_{\text{svd}} = \alpha_{\text{svd}}\, v + \sqrt{\frac{1 - \beta_{\text{svd}}^2}{n}}\, g + \Delta_v,$$

where Δ_u *and* Δ_v *contain the higher order terms, and* $g \sim \mathcal{CN}(0, I)$. *We conjecture that* $||\Delta_u||_2 \xrightarrow{\text{a.s.}} 0$ *and* $||\Delta_v||_2 \xrightarrow{\text{a.s.}} 0$ *as* $m \to \infty$.

Assume that Conjectures IV.1 and IV.2 hold. Then, for large enough m, we may use the approximations

$$\widehat{u}_{\text{svd}} \approx \alpha_{\text{svd}}\, u + \sqrt{\frac{1 - \alpha_{\text{svd}}^2}{m}}\, g, \tag{4.15a}$$

$$\widehat{v}_{\text{svd}} \approx \beta_{\text{svd}}\, v + \sqrt{\frac{1 - \beta_{\text{svd}}^2}{n}}\, g. \tag{4.15b}$$

Claim IV.1. *Assume that the hypotheses of Proposition IV.1 hold, and that Conjectures IV.2 and IV.1 hold. Then, we have that, as* $m_1, m_2, m_3, n \to \infty$, $m_i/n \to c_i$ *for* $i \in \{1, 2, 3\}$,

$m_i m_j/n \to c_{ij}$ *for* $i,j \in \{1,2,3\}$, $m/n \to c_0$,

$$|\widehat{u}_i^H u|^2 = |\widehat{a}_i^H a_i|^2 \cdot |\widehat{b}_i^H b_i|^2 \cdot |\widehat{c}_i^H c_i|^2$$

where

$$|\widehat{a}_i^H a_i|^2,\ |\widehat{b}_i^H b_i|^2,\ |\widehat{c}_i^H c_i|^2 \to \textit{ quantities in Table 4.2}, \quad \textit{for } i \in \{KPCA3{,}1, \ldots, KPCA3{,}8\},$$

if

$$\sigma_0 > \max(\sigma_{\mathsf{crit,a,i}},\ \sigma_{\mathsf{crit,b,i}},\ \sigma_{\mathsf{crit,c,i}}) \textit{ from Table 4.3} \quad \textit{for } i \in \{KPCA3{,}1, \ldots, KPCA3{,}8\}.$$

In Table 4.2,

$$\begin{aligned}
T_{4c,1} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right]^2, \\
T_{4c,2} &= c_3(c_1 + \sigma_0^2)^2\left[\sigma_0^4 - c_0 + c_2(c_1 + \sigma_0^2)\right]^2, \\
T_{4c,3} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right], \\
T_{4c,4} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2 + c_3(c_1 + \sigma_0^2)\left[\sigma_0^4 - c_0 + c_2(c_1 + \sigma_0^2)\right]\right], \\
T_{5b,1} &= [(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2]^2, \\
T_{5b,2} &= c_2(c_3 + \sigma_0^2)^2[\sigma_0^4 - c + c_1(c_3 + \sigma_0^2)]^2, \\
T_{5b,3} &= [(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2], \\
T_{5b,4} &= [(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2 + c_2(c_3 + \sigma_0^2)[\sigma_0^4 - c_0 + c_1(c_3 + \sigma_0^2)]], \\
T_{6b,1} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right]^2, \\
T_{6b,2} &= c_2(c_1 + \sigma_0^2)^2\left[\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right]^2, \\
T_{6b,3} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right], \\
T_{6b,4} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2 + c_2(c_1 + \sigma_0^2)\left[\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right]\right], \\
T_{8a,1} &= \left[(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2\right]^2, \\
T_{8a,2} &= c_1(c_3 + \sigma_0^2)^2\left[c_2(c_3 + \sigma_0^2) + \sigma_0^4 - c_0\right]^2, \\
T_{8a,3} &= \left[(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2\right], \\
T_{8a,4} &= \left[(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2 + c_1(c_3 + \sigma_0^2)\left[\sigma_0^4 - c_0 + c_2(c_3 + \sigma_0^2)\right]\right],
\end{aligned}$$

and in Table 4.3, the quantities $\kappa_{4,c}$, $\kappa_{5,a}$, $\kappa_{6,b}$, *and* $\kappa_{8,a}$ *are the positive, real valued roots of the eight degree polynomials in Table 4.1.*

Proof. See Appendix D. □

Quantity in Table 4.3	Positive, real root of polynomial in σ
$\kappa_{4,c}$	$(\sigma^4 - c_0)^2 - c_{23}(c_1 + \sigma^2)^2 - \sqrt{c_3}(c_1 + \sigma^2)\left[c_2(c_1 + \sigma^2) + \sigma^4 - c_0\right] = 0$
$\kappa_{5,a}$	$(\sigma^4 - c_0)^2 - c_{12}(c_3 + \sigma^2)^2 - \sqrt{c_2}(c_3 + \sigma^2)\left[c_1(c_3 + \sigma^2) + \sigma^4 - c_0\right] = 0$
$\kappa_{6,b}$	$(\sigma^2 - c_0)^4 - c_{23}(c_1 + \sigma^2)^2 - \sqrt{c_2}(c_1 + \sigma^2)\left[c_3(c_1 + \sigma^2) + \sigma^4 - c_0\right] = 0$
$\kappa_{8,a}$	$(\sigma^4 - c_0)^2 - c_{12}(c_3 + \sigma^2)^2 - \sqrt{c_1}(c_3 + \sigma^2)\left[c_2(c_3 + \sigma^2) + \sigma^4 - c_0\right] = 0$

Table 4.1: Definition of quantities in Table 4.3

Algorithm	$\lvert\widehat{a}_i^H a\rvert^2$	$\lvert\widehat{b}_i^H b\rvert^2$	$\lvert\widehat{c}_i^H c\rvert^2$
KPCA3,1	1	1	1
KPCA3,2	1	1	1
KPCA3,3	$\frac{(\sigma_0 - c_0)^2 - c_0 c_3}{(\sigma_0 - c_0)^2}$	$\frac{(\sigma_0 - c_0)^2 - c_0 c_3^2}{(\sigma_0 - c_0)^2}$	$\frac{(\sigma_0^4 - c_0)^2 - c_3(\sigma_0^2 + c_{12})^2}{(\sigma_0^4 - c_0)(\sigma_0^4 - c_0 + c_3(\sigma_0^2 + c_{12}))}$
KPCA3,4	$\frac{\sigma_0^4 - c_0}{\sigma_0^2(\sigma_0^2 + c_1)}$	$\frac{(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2}{(\sigma_0^4 - c)\left(\sigma_0^4 - c_0 + c_2(c_1 + \sigma_0^2)\right)}$	$\frac{T_{4c,1} - if T_{4c,2}}{T_{4c,3} \cdot T_{4c,4}}$
KPCA3,5	$\frac{(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2}{(\sigma_0^4 - c_0)(\sigma_0^4 - c_0 + c_1(c_3 + \sigma_0^2))}$	$\frac{T_{5b,1} - T_{5b,2}}{T_{5b,3} \cdot T_{5b,4}}$	$\frac{\sigma_0^4 - c_0}{\sigma_0^2\left(\sigma_0^2 + c_3\right)}$
KPCA3,6	$\frac{\sigma_0^4 - c_0}{\sigma_0^2(\sigma_0^2 + c_1)}$	$\frac{T_{6b,1} - T_{6b,2}}{T_{6b,3} T_{6b,4}}$	$\frac{(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2}{(\sigma_0^4 - c)\left(\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right)}$
KPCA3,7	$\frac{(\sigma_0^4 - c_0)^2 - c_1(c_{23} + \sigma_0^2)}{(\sigma_0^4 - c_0)(\sigma_0^4 - c_0 + c_1(c_{23} + \sigma_0^2))}$	$\frac{(\sigma_0^4 - c_0)^2 - c_0 c_1}{(\sigma_0^4 - c_0)[\sigma_0^4 - c_0 + c_{12}]}$	$\frac{(\sigma_0^4 - c_0)^2 - c_0 c_1}{(\sigma_0^4 - c_0)[\sigma_0^4 - c_0 + c_{13}]}$
KPCA3,8	$\frac{T_{8a,1} - T_{8a,2}}{T_{8a,3} \cdot T_{8a,4}}$	$\frac{(\sigma_0^4 - c_0)^2 - c_{12}(c_3 + \sigma_0^2)^2}{(\sigma_0^4 - c_0)[\sigma_0^4 - c_0 + c_2(c_3 + \sigma_0^2)]}$	$\frac{\sigma_0^4 - c_0}{\sigma_0^2(\sigma_0^2 + c_3)}$

Table 4.2: Inner-product-squared estimation accuracy of KPCA3 algorithms.

Algorithm	$\sigma_{\mathsf{crit,a}}$	$\sigma_{\mathsf{crit,b}}$	$\sigma_{\mathsf{crit,c}}$
KPCA3,1	$c_0^{1/4}$	$c_0^{1/4}$	$c_0^{1/4}$
KPCA3,2	$c_0^{1/4}$	$c_0^{1/4}$	$c_0^{1/4}$
KPCA3,3	$c_0^{1/4}$	$c_0^{1/4}$	$(c_0 + \sqrt{c_3})^{1/4}$
KPCA3,4	$c_0^{1/4}$	$\left(\frac{\sqrt{c_{23}} + \sqrt{c_{23} + 4c_0}}{2}\right)^{1/2}$	$\kappa_{4,c}$
KPCA3,5	$\kappa_{5,a}$	$\left(\frac{\sqrt{c_{12}} + \sqrt{c_{12} + 4c_0 + 4\sqrt{c_{12}}c_3}}{2}\right)^{1/2}$	$c_0^{1/4}$
KPCA3,6	$c_0^{1/4}$	$\kappa_{6,b}$	$\left(\frac{\sqrt{c_{23}} + \sqrt{c_2 3 + 4c + 4\sqrt{c_{23}}c_1}}{2}\right)^{1/2}$
KPCA3,7	$\left(\frac{\sqrt{c_1} + \sqrt{c_1 + 4c_0 + 4\sqrt{c_1}c_{23}}}{2}\right)^{1/2}$	$(c_0 + \sqrt{c_0}\sqrt{c_{12}})^{1/4}$	$(c_0 + \sqrt{c_0}\sqrt{c_{12}})^{1/4}$
KPCA3,8	$\kappa_{8,a}$	$\left(\frac{\sqrt{c_{12} + \sqrt{c_{12} + 4c_0 + 4\sqrt{c_{12}}c_3}}}{2}\right)^{1/2}$	$c_0^{1/4}$

Table 4.3: Critical SNR for informative estimation in KPCA3 algorithms.

IV.5 Numerical Simulations

Figure 4.1 plots the estimation performance $|\langle \widehat{u}, u \rangle|^2$, where $\widehat{u}$ is estimated using all the KPCA3 algorithms. The discrete points were generated over 2000 Monte Carlo simulations, for a system with $m_1 = 10, m_2 = 9, m_3 = 8$ and $n = 800$. We see that all KPCA algorithms outperform the SVD. Among the KPCA algorithms, KPCA3, 1 and KPCA3, 2, which first compute the SVD estimate of $\widehat{u}_{\mathsf{svd}}$ and then fit a Kronecker structure to it outperform the other KPCA algorithms. The continuous lines are plots of our analytical prediction of performance. We see that the analytical prediction closely matches the empirical performance.

IV.5.1 Application: MIMO STAP

In Multiple-Input-Multiple-Output Space-Time-Adaptive-Processing (MIMO STAP), the vectors a, b and c take the form [17, 69]

$$a(f_t) = \frac{1}{\sqrt{n_1}} \begin{bmatrix} 1 & e^{j2\pi f_t} & \dots & e^{j2\pi(n_1-1)f_t} \end{bmatrix}^T \tag{4.16a}$$

$$b(f_r) = \frac{1}{\sqrt{n_2}} \begin{bmatrix} 1 & e^{j2\pi f_r} & \dots & e^{j2\pi(n_2-1)f_r} \end{bmatrix}^T \tag{4.16b}$$

$$c(f_d) = \frac{1}{\sqrt{n_3}} \begin{bmatrix} 1 & e^{j2\pi f_d} & \dots & e^{j2\pi(n_3-1)f_d} \end{bmatrix}^T. \tag{4.16c}$$

Here $j = \sqrt{-1}$, and f_t, f_r and f_d are the normalized spatial frequencies at the transmitter and receiver, and normalized Doppler frequency respectively, given by

$$f_r = \frac{d_r}{\lambda}\cos(\theta_r),\ f_t = \frac{d_t}{\lambda}\cos(\theta_t),\ \text{and}\ f_d = \frac{v_r\, T_p}{d_r} f_r + \frac{v_t\, T_p}{d_t} f_t. \tag{4.17}$$

Here, θ_t and θ_r denote the directions of a narrowband source with respect to the transmitter and receiver uniform linear arrays, which have elements spaced d_t and d_r apart, respectively. Furthermore, λ is the the wavelength, T_p is the coherent pulse interval and v_r, v_t are the speeds of clutter or target relative to the receiver and transmitter. In MIMO STAP, we obtain Y as in (4.1), The DMR beamformer has the form [70, 67, 71]

$$\widehat{w} = \frac{(I + \sigma_0^2 \widehat{u}\widehat{u}^H)^{-1}\, u_t}{u_t^H (I + \sigma_0^2 \widehat{u}\widehat{u}^H)^{-1} u_t}, \tag{4.18}$$

where u_t is the manifold vectors associated with the desired target, and $\widehat{u}$ is an estimate of u_i.

In Figure 4.2, we demonstrate the application to MIMO STAP. We consider a system with dimensions as in Fig. 4.1, for an MTI system with wavelength $\lambda = 0.67$ m, array elements spaced half wavelength apart. The target has velocity of $v_t = 20$ m/s and $v_r = 30$ m/s with respect to the receiver, while interferer is stationary. The pulse repetition frequency is $1/T_p = 600$ Hz . We plot the notch depth, defined as

$$\eta := |\widehat{w}^H u(\theta_{t,i}, \theta_{r,i}, f_{d,i})|^2.$$

Here $\theta_{t,i}$, $\theta_{r,i}$ and $f_{d,i}$ are the angles and Dopper shifts due to the interferer, as in (4.17), $\widehat{w}$ is computed using (4.18) from $\widehat{u}$ is found using the KPCA3 algorithms. We see that all KPCA algorithms suppress interference better than the canonical SVD estimate. Among the KPCA algorithms, KPCA3, 1 and KPCA3, 2 best suppress the interference as expected from Fig. 4.1.

In Fig. 4.3 and 4.4, we plot the beampattern of a MIMO STAP system, defined as

$$\mathrm{BP}(\theta_t, \theta_r, f_d) = |\widehat{w}^H u(\theta_t, \theta_r, f_d)|^2. \tag{4.19}$$

The receiver beampattern in plotted in Fig. 4.3 and the transmitter beampattern in Fig. 4.4, for a system with array dimensions as in Fig. 4.1, INR 5dB, for a target located at $\theta_{t,s} = 45^o, \theta_{r,s} = 10^o$ and an interferer located at $\theta_{t,i} = 20^o, \theta_{r,i} = 40^o$. We find that using KPCA3, 1 better suppresses interference relative to the SVD. For clarity of plots within the figure, we have plotted only the beampattern of KPCA3, 1, but using any of the KPCA, 3 algorithms will yield an improvement in performance, as seen from Figs. 4.1 and 4.2.

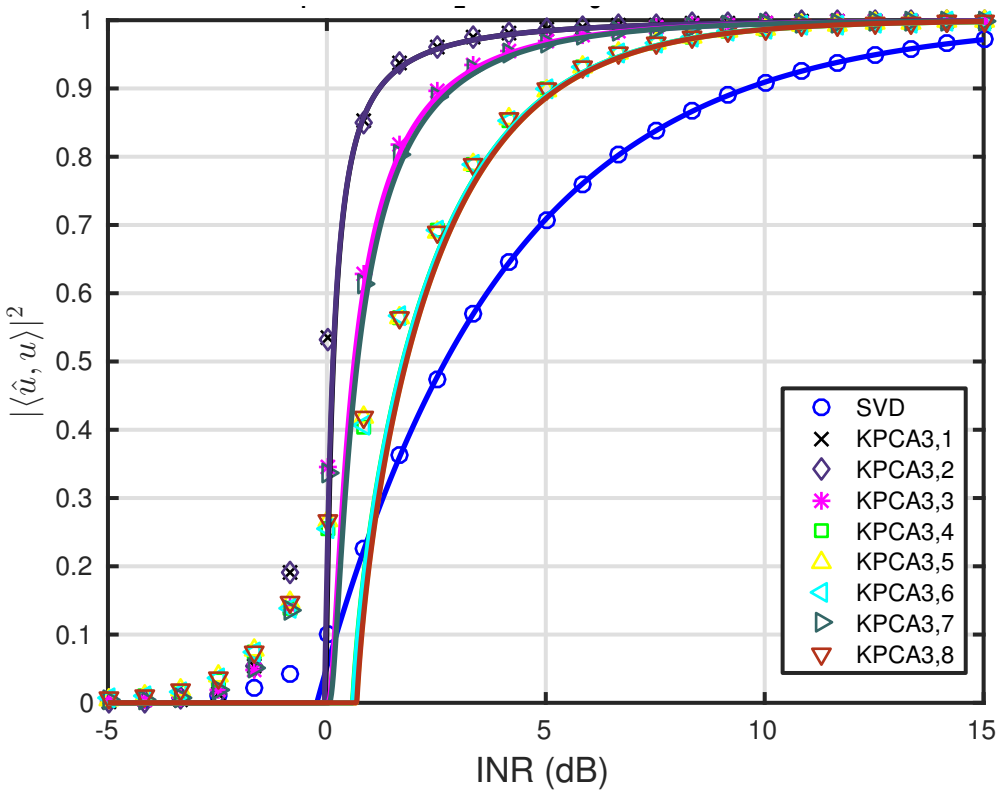


Figure 4.1: Plot of estimation performance $|\widehat{u}^H u|^2$ vs INR (dB), for a system with $m_1 = 10, m_2 = 9, m_3 = 8$ and $n = 800$ in (4.1), where $\widehat{u} \in \{\widehat{u}_{\mathsf{svd}}, \widehat{u}_{\mathsf{kpca3,1}}, \ldots, \widehat{u}_{\mathsf{kpca3,8}}\}$. The estimation performance of all KPCA3 algorithms is superior to the canonical SVD estimate. In particular, KPCA3,1 and KPCA3,2 which first compute $\widehat{u}_{\mathsf{svd}}$ and then fit a Kronecker structure to it, give the best performance over KPCA3,3 and KPCA3,8 which exploit the Kronecker structure directly on the observed data. The solid line plots our analytical prediction of estimation performance. We see that our analytical prediction closely matches the empirical performance.

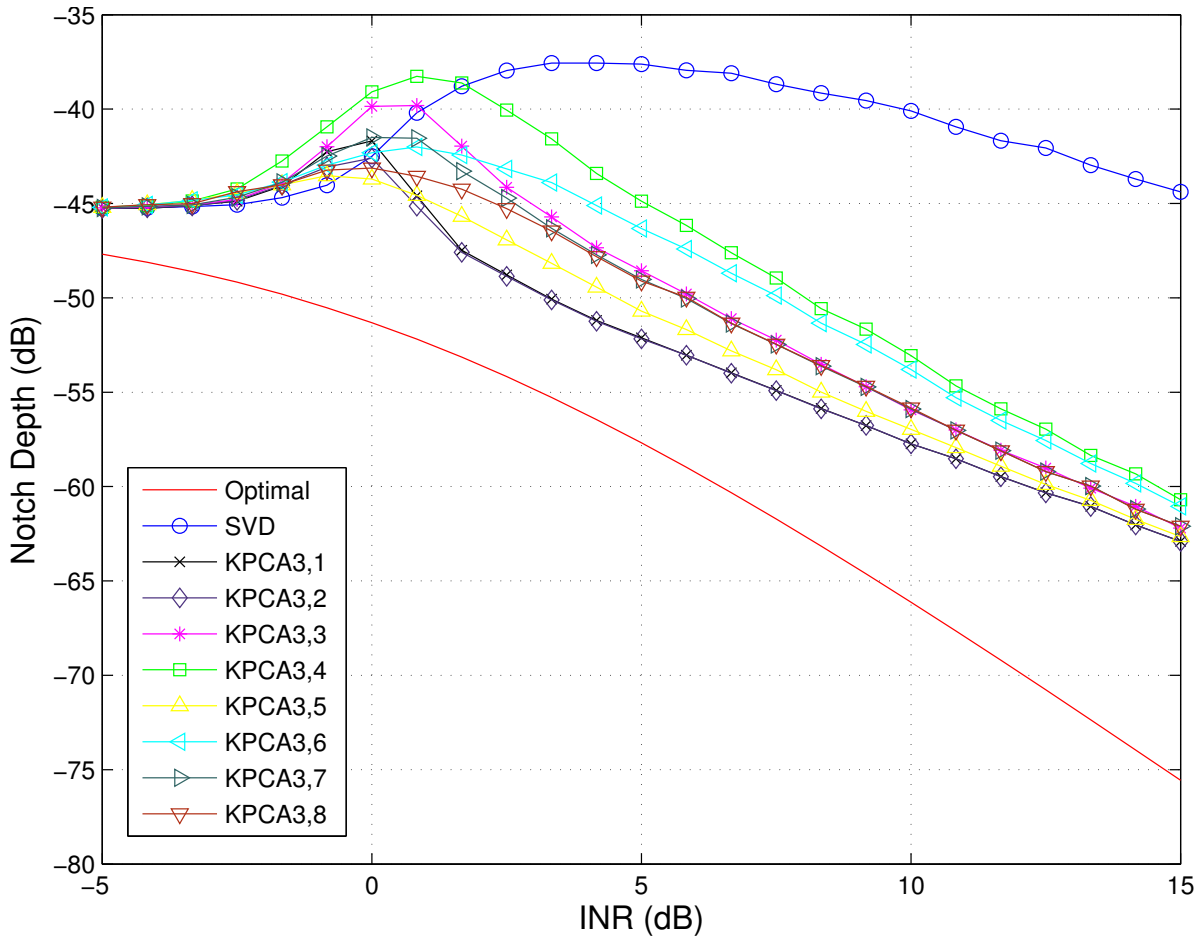


Figure 4.2: Plot of Notch Depth (dB) at the interferer for a MIMO STAP system, with the same dimensions as in Fig. 4.1. Computing the MVDR filter using KPCA algorithms gives a deeper notch depth compared to forming the filter using the canonical estimate $\widehat{u}_{\mathsf{svd}}$. At high INRs, we get a notch depth gain of around 10 dB using the new algorithms. Among the KPCA algorithms, KPCA3, 1 and KPCA3, 2 give the deepest notch depths, as we expect from the plot in Fig. 4.1.

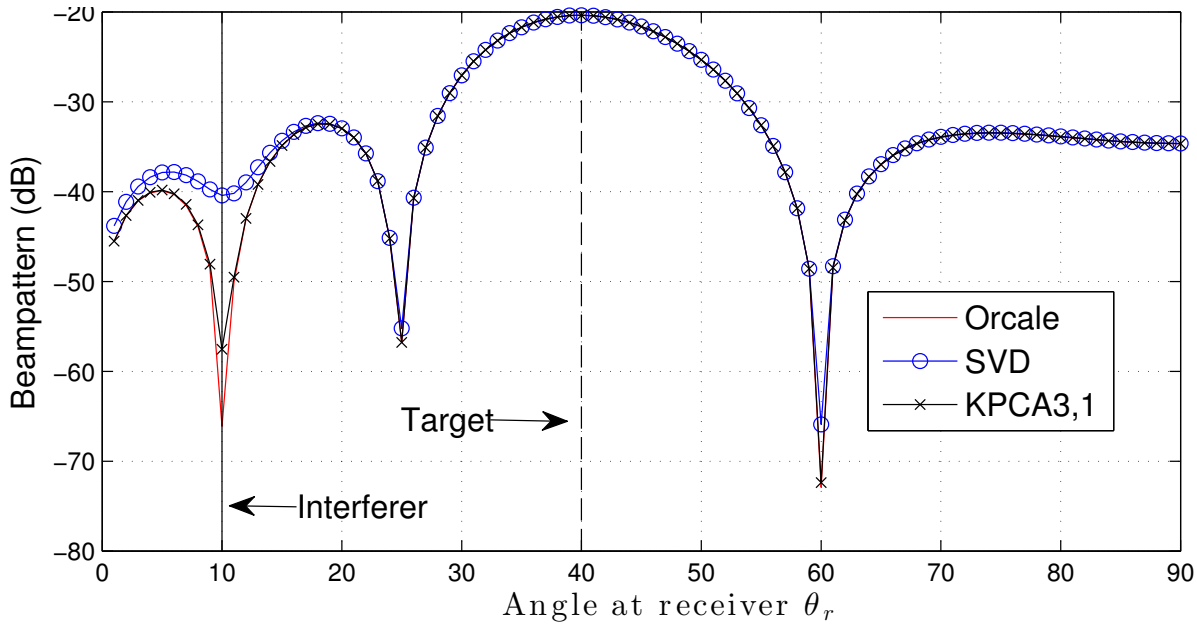


Figure 4.3: Beampattern $\mathrm{BP}(\theta_r, \theta_{t,s}, f_{d,s})$ as in (4.19) for the system from Fig. 4.1, plotted against θ_r (in degrees) at true values of $\theta_{t,s} = 20^o$ and $f_{d,s}$ as in (4.17). The target is at 40^o relative to the receiver and interferer is at 10^o. The INR was fixed at 5 dB. The oracle beampattern was computed by using the population manifold vector of the clutter to compute the MVDR beamformer, i.e. $\widehat{u} = u_i$ in (4.18). Similarly, the SVD and KPCA3, 1 plots corresponding to using $\widehat{u} = \widehat{u}_{\mathrm{svd}}$ and $\widehat{u} = \widehat{u}_{\mathrm{kpca3,1}}$ in (4.18). We see that using the KPCA3, 1 estimate gives a deeper notch depth at the interferer location, relative to using the SVD estimate.

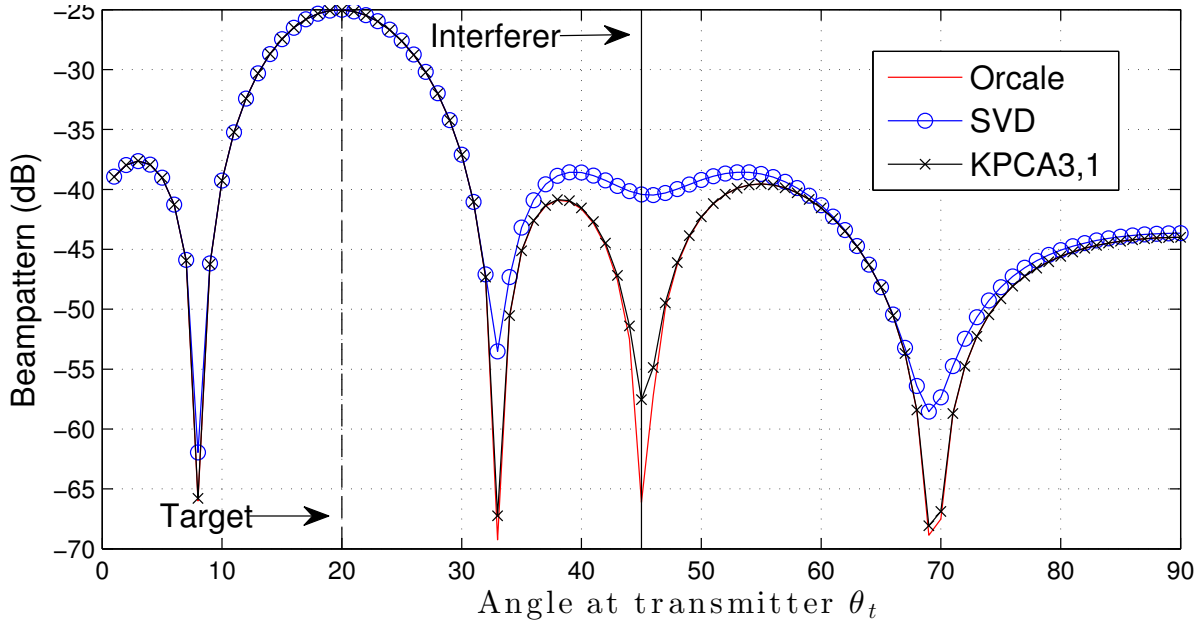


Figure 4.4: Beampattern $\mathrm{BP}(\theta_{r,s}, \theta_t, f_{d,s})$ as in (4.19) for the system from Fig. 4.1, plotted against θ_t (in degrees) at true $\theta_r = 40^o$ and $f_{d,s}$ as in (4.17). In this plot target is at 20^o relative to the transmitter and interferer is at 10^o. This plot is the analogue of Fig. 4.3 at the transmitter. Once again, we see a deeper notch depth in the direction of the interferer when we use $\widehat{u} = \widehat{u}_{\mathrm{kpca3,1}}$ relative to $\widehat{u} = u_{\mathrm{svd}}$ in (4.18).

IV.6 Conclusions

We proposed eight algorithms for estimating singular vectors with Kronecker product structure. We provided analytical expressions that quantify their relative estimation performance. We showed using simulations that they all provide better estimation performance than the SVD, and demonstrated better interferer suppression in a MIMO STAP system.

CHAPTER V

BeamShrink: An Algorithm for Improved Weighting of SVD Based Beamformers

V.1 Introduction

Beamforming via sensor arrays is an important task in many signal processing applications. It is central to direction of arrival estimation [3, 27, 1], space time adaptive processing (STAP) [55, 16], MIMO STAP [17, 69], acoustic vector sensor array processing [72, 73], *etc.*

A critical step in the beamforming process is the estimation of the covariance matrix of data. Robust estimation of covariance for beamforming is an active area of research; see, for *e.g.* [74, 75, 76, 77]. In [78], the authors considered the problem of covariance estimation from Gaussian samples, and proposed shrinkage algorithms for improved covariance matrix estimation.

The sample covariance matrix is a poor choice for covariance estimation in array processing applications, due to the limited number of samples, and due to robustness considerations. A popular choice of covariance matrix estimate is the SVD based estimate of the covariance matrix [79]. This estimate is based on the observation that the underlying clutter signal subspace is low rank relative to the dimensions of the system and the number of samples. It takes as its input the $m \times n$ data matrix, computes its principal r singular values and vectors, and returns $\widehat{R}_{\mathsf{svd}}$, an estimate of the underlying covariance matrix R. Here r is the rank of the underlying signal subspace, which corresponds to the number of clutter sources in the system.

The covariance matrix estimate $\widehat{R}_{\text{svd}}$ has the form

$$\widehat{R}_{\text{svd}} = I + \sum_{i=1}^{r} \widehat{\sigma}^2_{\text{svd,i}}\, \widehat{u}_{\text{svd,i}}\, \widehat{u}^H_{\text{svd,i}}.$$

The matrix $\widehat{u}_{\text{svd,i}}\, \widehat{u}^H_{\text{svd,i}}$ may be interpreted as a projection matrix onto (a subspace of) the clutter subspace, while $\widehat{\sigma}_{\text{svd,i}}$ functions as a *weight* associated with this projection matrix.

In this chapter, we focus on the dominant mode rejection (DMR) beamformer for clutter supression [67, 71]. The ideal DMR beamformer is a function of R^{-1}, the inverse of the population covariance matrix of the data. Canonically, the beamformer is computed using $\widehat{R}^{-1}_{\text{svd}}$, the inverse of the SVD based covariance matrix estimate. In general, we expect that any estimate of the inverse covariance matrix which is closer to R^{-1} will yield better beamformer performance. Accordingly, we consider an optimization problem whose objective is to minimize the difference

$$\|R^{-1} - \widehat{R}^{-1}\|_F^2.$$

Minimizing this objective function is a challenging problem in general. However, inspired by the work in [15], we begin by retaining the top r singular vectors, and the resulting outerproduct matrices, in the covariance matrix estimate; we allow the weights (which were previously functions of the singular values of the data) to vary. This results in the objective function

$$\|R^{-1} - I + \sum_{i=1}^{r} z_i^2\, \widehat{u}_{\text{svd,i}}\, \widehat{u}^H_{\text{svd,i}}\|_F^2, \tag{5.1}$$

where $\{z_i\}_{i=1}^r$ are the variables we are minimizing over.

The solution to this optimization problem, and the nature of the optimum form the core of this chapter.

We will see that we can find an exact, closed form minimum of the objective function in (5.1). This solution will be a function of $\{\sigma_i\}_{i=1}^r$, the r singular values of the clutter subspace, and $\{|\widehat{u}^H_{\text{svd,i}} u_i|^2\}_{i=1}^r$, the inner-product-squared between the i^{th} sample left singular vector of the data, and the corresponding underlying (noise-free) singular vector.

As with R, we cannot observe $\{\sigma_i\}_{i=1}^r$ or $\{|\widehat{u}^H_{\text{svd,i}} u_i|^2\}_{i=1}^r$. However, we will see that we can exactly characterize the asymptotic behavior of these quantities as $m, n \to \infty$, $m/n \to c$, using recent results from random matrix theory. These limiting quantities are integral transforms of the limiting singular value distribution of the noise only matrix, and can be computed analytically for a large class of noise distributions.

The work in this chapter goes above and beyond these classes of distributions for which we can analytically characterize the solution of (5.1). We present a data driven algorithm, called BeamShrink, using which we can consistently estimate the solution of (5.1) in a data driven manner. In this way, we can estimate the optimum weights for inverse covariance estimation for more general classes of noise distributions.

We will see that the solution to (5.1) defines a non-convex *shrinkage* operator on the singular values of the clutter subspace. In other words, the optimum weights for inverse covariance matrix estimation are not the singular values themselves, but shrinkage functions of the singular values. The shrinkage nature of the optimum weights arises due to two reasons: a. the sample singular values of clutter subspace computed from the data are biased upwards relative to the true clutter singular values, due to the additive noise; and b. the sample singular vectors are biased estimates of the underlying singular vectors.

Furthermore, we will also see that the optimum weights derived in this chapter exhibit a phase transition phenomenon, inherited from a phase transition behavior in the sample singular vectors. If the clutter singular value is above a critical value, which is a simple function of the system dimensions and sample size, the corresponding sample singular vector lies on a cone around the underlying singular vector, and is correlated with it. However, if the singular value is below the same critical value, then the sample singular vector is uncorrelated with the underlying singular vector and essentially random.

A pleasant consequence of being able to estimate these shrinkage weights from the data is robustness to rank overestimation. If the exact rank of the underlying low rank signal is unknown, and it is overestimated, then the covariance matrix estimate will involve extraneous, uninformative components. Since the weights output by the BeamShrink algorithm are functions of singular vector informativeness (in terms of inner-product-squared), we are in a position to identify uninformative components in a data driven manner. The uninformative components are shrunk more in forming the covariance matrix estimate, and thus rank overestimation is mitigated.

The rest of this chapter is organized as follows. We describe the signal model and beamformer in Section V.2. We derive the optimum weight for inverse covariance matrix estimation of rank one signals in Section V.3. We validate our analysis with numerical simulations in Section V.5, and concluding remarks are in Section V.6.

V.2 Data Model and Beamformer

Suppose we observe data matrix $Y \in \mathbb{C}^{m \times n}$, which has the form:

$$Y = \sum_{i=1}^{r} \sigma_{c,i}\, u_{c,i}\, s_i^H + Z, \tag{5.2}$$

where $\{\sigma_{c,i}\}_{i=1}^r \in \mathbb{R}_+$ are the clutter powers, $\{u_{c,i}\}_{i=1}^r \in \mathbb{C}^m$ are the (unit norm) steering vectors associated with clutter locations, $\{s_i\}_{i=1}^r \in \mathbb{C}^n$ are the (unit norm) signal vectors, and Z is the noise matrix, independent of the signal, whose elements have 0 mean and covariance $1/n$.

Let

$$u_t \in \mathbb{C}^m \tag{5.3}$$

be the known steering vector associated with a *target*. The optimal Dominant Mode Rejection (DMR) beamformer that rejects clutter at u_c and steers the array towards the target u_t is given by [71, 80]:

$$w_{\text{opt}} = \frac{R^{-1} u_t}{\|R^{-1} u_t\|_2} \tag{5.4}$$

where $R := \mathbb{E}\left[YY^H\right]$ is the population covariance matrix of the data.

Let

$$U_c := \begin{bmatrix} u_{c,1} & \dots & u_{c,r} \end{bmatrix}, \quad \Sigma_c := \operatorname{diag}(\sigma_{c,1}, \dots, \sigma_{c,r}).$$

Then R is given by

$$\begin{aligned} R &= I + \sum_{i=1}^{r} \sigma_{c,i}^2\, u_{c,i}\, u_{c,i}^H \\ &= I + U_c\, \Sigma_c^2\, U_c^H. \end{aligned} \tag{5.5}$$

Since $U_c\, \Sigma_c^2\, U_c^H$ is a Hermetian matrix, it has an eigenvalue decomposition of the form

$$U_c\, \Sigma_c^2\, U_c^H = U_s\, \Sigma_s^2\, U_s^H,$$

where

$$U_s = \begin{bmatrix} u_1 & \dots & u_r \end{bmatrix}, \quad \Sigma_s = \operatorname{diag}(\sigma_1, \dots, \sigma_r), \tag{5.6}$$

are the eigenvectors and eigenvalues, respectively, of $U_c\, \Sigma_c^2\, U_c^H$. Using (5.6) in (5.5), we have

that

$$R = I + U_s \Sigma_s^2 U_s^H. \tag{5.7}$$

In practice, we do not know R since observed data is noisy. However, we can estimate it by leveraging the fact that the clutter signal lies in a rank r subspace of the data. Canonically, R is estimated via the SVD of Y, as follows:

$$\widehat{R}_{\text{svd}} = I + \sum_{i=1}^{r} (\widehat{\sigma}_{\text{svd,i}}^2 - 1)\, \widehat{u}_{\text{svd,i}}\, \widehat{u}_{\text{svd,i}}^H, \tag{5.8}$$

where $\widehat{u}_{\text{svd,i}}$ is the i^{th} principal left singular vector of Y, corresponding to its i^{th} largest singular value, $\widehat{\sigma}_{\text{svd,i}}$. In (5.8), we may interpret $\widehat{u}_{\text{svd,i}}\, \widehat{u}_{\text{svd,i}}^H$ as the estimate of the projection matrix onto a clutter subspace, and

$$\widehat{z}_{\text{svd,i}} := \sqrt{\widehat{\sigma}_{\text{svd,i}}^2 - 1} \tag{5.9}$$

as the *weight* associated with this estimate. The SVD based beamformer associated with this estimate is given by

$$\widehat{w}_{\text{svd}} = \frac{\widehat{R}_{\text{svd}}^{-1} u_t}{\|\widehat{R}_{\text{svd}}^{-1} u_t\|_2}. \tag{5.10}$$

The focus of this chapter is to derive new weights that yield a better estimate of the inverse covariance matrix, in a sense that will be made precise in Section V.3.

V.3 Optimum Weights for Inverse Covariance Estimation

The beamformer of interest to us is a function of the inverse covariance matrix estimate (see (5.4), (5.10)). We expect that an improved estimate of the inverse covariance matrix will lead to improved beamformer performance. To that end, let

$$\widehat{R}_z := I + \sum_{i=1}^{r} z_i^2\, \widehat{u}_{\text{svd,i}}\, \widehat{u}_{\text{svd,i}}^H. \tag{5.11}$$

The covariance matrix estimate $\widehat{R}_z$ is a function of $\{\widehat{u}_{\text{svd,i}}\}_{i=1}^r$, the singular vector estimates returned from the SVD of Y, and (in place of $\widehat{z}_{\text{svd,i}}$) some arbitrary weights $\{z_i\}_{i=1}^r$. We

formulate the optimization problem

$$z_{\mathsf{bsh}} = \underset{z \in \mathbb{R}^r}{\arg\min} \, \|R^{-1} - \widehat{R}_z^{-1}\|_F^2, \tag{5.12}$$

where R^{-1} is the inverse of the population covariance matrix given by (5.7). Using the weights z_{bsh} in place of $\widehat{z}_{\mathsf{svd}}$ in (5.9) will yield an estimate of the inverse covariance matrix which is closer to the inverse of the population covariance matrix, relative to the SVD based estimate, in the Frobenius norm sense. We term the weights $\{z_{\mathsf{bsh,i}}\}_{i=1}^r$ as "BeamShrink" weights, since they will given by a shrinkage operation on $\{\sigma_i\}_{i=1}^r$.

In this section, we determine the BeamShrink weights $\{z_{\mathsf{bsh,i}}\}_{i=1}^r$, and propose a data driven algorithm to estimate them from Y. The following theorem gives us an exact, deterministic solution to the optimization problem in (5.12).

Theorem V.1. *Let Y be the observed data as in (5.2), whose population covariance matrix R is given by (5.5). Let $\widehat{R}_{\mathsf{svd}}$, given by (5.8), be the SVD based estimate of R. Let $\{u_i, \sigma_i\}_{i=1}^r$ be as defined in (5.6). Let $\{z_{\mathsf{bsh,i}}\}_{i=1}^r$ be the solution to the optimization problem*

$$z_{\mathsf{bsh}} = \underset{z \in \mathbb{R}^r}{\arg\min} \, \|R^{-1} - \widehat{R}_z^{-1}\|_F^2$$

as in (5.12). Then, we have that, for $i = 1, \ldots, r$,

$$z_{\mathsf{bsh,i}} = \sqrt{\frac{K_{ii}}{1 - K_{ii}}} \tag{5.13}$$

where

$$K_{ii} = \sum_{j=1}^{r} \frac{\sigma_j^2}{1 + \sigma_j^2} \, |\widehat{u}_{\mathsf{svd,i}}^H \, u_j|^2.$$

To prove this theorem, we will require the following lemmas.

Lemma V.1. *Let $X \in \mathbb{C}^{m \times n}$ and $r < \min(m, n)$. Let D_{opt} solve the optimization problem*

$$D_{\mathsf{opt}} = \underset{D := \mathrm{diag}(d_1, \ldots, d_r, 0, \ldots, 0) \in \mathbb{R}_+^{m \times n}}{\arg\min} \|X - D\|_F^2.$$

Then, for $i = 1, \ldots, r$,

$$(D_{\mathsf{opt}})_{ii} = \max\left(0, \Re(X_{ii})\right).$$

Proof. The proof is given by [15, Lemma 4.1]. We reproduce it here for completeness. We

have that

$$\|X - D\|_F^2 = \sum_{i=1}^{r}(X_{ii} - d_i)^2 + \sum_{i=r+1}^{\min(m,n)} X_{ii}^2 + \sum_{i\neq j} X_{ij}^2 \geq \sum_{i=r+1}^{\min(m,n)} X_{ii}^2 + \sum_{i\neq j} X_{ij}^2.$$

The lower bound of error is achieved by setting $d_i = X_{ii}$ for $i = 1, \ldots, r$. The constraint $d_i \in \mathbb{R}_+$ is satisfied by projecting the X_{ii} onto $\mathbb{R}_+$, which gives us $\max(0, \mathfrak{R}(X_{ii}))$. □

Lemma V.2. *Let R, U_s, and Σ_s be as defined in (5.7), and let*

$$U := \begin{bmatrix} U_s & U_n \end{bmatrix}, \tag{5.14}$$

where U_n is an orthonormal basis for the orthocomplement of the range of U_s. Then,

$$R^{-1} = I - U \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^H$$

where

$$\operatorname{diag}\left(\frac{\sigma_i^2}{1+\sigma_i^2}\right) := \begin{bmatrix} \dfrac{\sigma_1^2}{1+\sigma_1^2} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \dfrac{\sigma_r^2}{1+\sigma_r^2} \end{bmatrix}.$$

Proof. Recall that, from the definition of R in (5.7),

$$R = I + U_s \Sigma_s^2 U_s^H.$$

Using the fact that U from (5.14) is an orthogonal matrix, we have that

$$R = UU^H + U \begin{bmatrix} \Sigma_s^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^H = U \begin{bmatrix} I + \Sigma_s^2 & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H.$$

Therefore,

$$
\begin{aligned}
R^{-1} &= U \begin{bmatrix} \operatorname{diag}\left(\dfrac{1}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H \\
&= U \begin{bmatrix} \operatorname{diag}\left(1 + \dfrac{1}{1+\sigma_i^2} - 1\right) & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H \\
&= U \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H - U \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H \\
&= I - U \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} U^H,
\end{aligned}
$$

which is the required result. □

Corollary V.1. *Let $\widehat{R}_z$ be as in (5.11), and define*

$$
\widehat{U}_s := \begin{bmatrix} \widehat{u}_{\mathsf{svd},1} & \dots & \widehat{u}_{\mathsf{svd},\mathsf{r}} \end{bmatrix}, \quad \widehat{U}_n := \begin{bmatrix} \widehat{u}_{\mathsf{svd},\mathsf{r}+1} & \dots & \widehat{u}_{\mathsf{svd},\mathsf{m}} \end{bmatrix}, \quad \widehat{U} := \begin{bmatrix} \widehat{U}_s & \widehat{U}_n \end{bmatrix}. \tag{5.15}
$$

Then, we have that

$$
\widehat{R}_z^{-1} = I - \widehat{U} \begin{bmatrix} \operatorname{diag}\left(\dfrac{z_i^2}{1+z_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \widehat{U}^H.
$$

Armed with the above results, we now prove Theorem V.1.

Proof of Theorem V.1. We wish to solve the optimization problem

$$
z_{\mathsf{bsh}} = \arg\min_{z \in \mathbb{R}_+^r} \|R^{-1} - \widehat{R}_z^{-1}\|_F^2.
$$

Using Lemma V.2 and Corollary V.1,

$$z_{\mathsf{bsh}} = \arg\min_{z\in\mathbb{R}^r_+} \left\| U \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^H - \widehat{U} \begin{bmatrix} \operatorname{diag}\left(\dfrac{z_i^2}{1+z_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \widehat{U}^H \right\|_F^2$$

$$= \arg\min_{z\in\mathbb{R}^r_+} \left\| \underbrace{\widehat{U}^H U \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U^H \widehat{U}}_{:=K} - \begin{bmatrix} \operatorname{diag}\left(\dfrac{z_i^2}{1+z_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right\|_F^2, \tag{5.16}$$

since Frobenius norm is unitarily invariant. Consider the matrix K from (5.16). Partitioning U and $\widehat{U}$ into top r and remaining columns,

$$K = \begin{bmatrix} \widehat{U}_s^H U_s & \widehat{U}_s^H U_n \\ \widehat{U}_n^H U_s & \widehat{U}_n^H U_n \end{bmatrix} \begin{bmatrix} \operatorname{diag}\left(\dfrac{\sigma_i^2}{1+\sigma_i^2}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} U_s^H \widehat{U}_s & U_s^H \widehat{U}_n \\ U_n^H \widehat{U}_s & U_n^H \widehat{U}_s \end{bmatrix}$$

$$= \begin{bmatrix} \widehat{u}_{\mathsf{svd},1}^H u_1 & \dots & \widehat{u}_{\mathsf{svd},1}^H u_r \\ & \ddots & \\ \widehat{u}_{\mathsf{svd},\mathsf{r}}^H u_1 & \dots & \widehat{u}_{\mathsf{svd},\mathsf{r}}^H u_r \end{bmatrix} \begin{bmatrix} \dfrac{\sigma_1^2}{1+\sigma_1^2} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \dfrac{\sigma_r^2}{1+\sigma_r^2} \end{bmatrix} \begin{bmatrix} u_1^H \widehat{u}_{\mathsf{svd},1} & \dots & u_1^H \widehat{u}_{\mathsf{svd},\mathsf{r}} \\ & \ddots & \\ u_r^H \widehat{u}_{\mathsf{svd},1} & \dots & u_r^H \widehat{u}_{\mathsf{svd},\mathsf{r}} \end{bmatrix}$$

$$= \sum_{j=1}^{r} \frac{\sigma_j^2}{1+\sigma_j^2} \begin{bmatrix} \widehat{u}_{\mathsf{svd},1}^H u_j \\ \vdots \\ \widehat{u}_{\mathsf{svd},\mathsf{r}}^H u_j \end{bmatrix} \begin{bmatrix} u_j^H \widehat{u}_{\mathsf{svd},1} & \dots & u_j^H \widehat{u}_{\mathsf{svd},\mathsf{r}} \end{bmatrix}$$

$$= \sum_{j=1}^{r} \begin{bmatrix} \dfrac{\sigma_j^2}{1+\sigma_j^2} (\widehat{u}_{\mathsf{svd},1}^H u_j)(u_j^H \widehat{u}_{\mathsf{svd},1}) & * & * \\ * & \ddots & * \\ * & * & \dfrac{\sigma_j^2}{1+\sigma_j^2} (\widehat{u}_{\mathsf{svd},1}^H u_r)(u_r^H \widehat{u}_{\mathsf{svd},\mathsf{r}}) \end{bmatrix}.$$

Therefore, the diagonal elements of K are, for $i = 1, \ldots, r$,

$$K_{ii} = \sum_{j=1}^{r} \frac{\sigma_j^2}{1+\sigma_j^2} |\widehat{u}_{\mathsf{svd,i}}^H u_j|^2 \tag{5.17}$$

Now we use Lemma V.1 and (5.17) in (5.16). We have that the solution to the optimization problem in (5.12) is given by

$$\frac{z_{\mathsf{bsh,i}}^2}{1+z_{\mathsf{bsh,i}}^2} = \max(0, \Re\{K_{ii}\}) = K_{ii}, \tag{5.18}$$

since it is easily seen from (5.17) that K_{ii} are non-negative and real valued. Therefore, we have that

$$z_{\mathsf{bsh,i}} = \sqrt{\frac{K_{ii}}{1-K_{ii}}}$$

which is the required result. □

The BeamShrink weights $z_{\mathsf{bsh,i}}$ in (5.13) are functions of the singular values $\{\sigma_i\}_{i=1}^r$ and singular vectors $\{u_i\}_{i=1}^r$ through the inner-product-squared terms $|\widehat{u}_{\mathsf{svd,i}}^H u_j|^2$ in (5.13). Even thought we cannot observe these quantities, we can characterize them through exact, deterministic objects in the asymptotic limit as $m, n \to \infty$, $m/n \to c$, using recent results from random matrix theory. We do so below.

Assume $m \leq n$, without loss of generality. Let $\sigma_1(Z) \geq \sigma_2(Z) \geq \ldots \sigma_m(Z)$ be the ordered singular values of the noise matrix Z. Let μ_{Z_m} be the empirical singular value distribution, defined as

$$\mu_{Z_m} = \frac{1}{m} \sum_{i=1}^{m} \delta_{\sigma_i(Z)}.$$

Since Z has i.i.d., mean 0, variance $1/m$ entries, we have that [81] as $m, n \to \infty$,

$$\mu_{Z_m} \xrightarrow{\text{a.s.}} \mu_Z. \tag{5.19}$$

Here $\xrightarrow{\text{a.s.}}$ denotes almost sure weak convergence, and μ_Z is a non random probability measure compactly supported on $[a, b]$. Furthermore, it is known that $\sigma_1(Z) \xrightarrow{\text{a.s.}} b$.

The following theorem characterizes the asymptotic behavior of the BeamShrink weights, in the large array and large sample limit, *i.e.*, as $m, n \to \infty$, $m/n \to c$.

Theorem V.2. *Let Y be data observed as in (5.2). Let $\{\widehat{u}_{\mathsf{svd,i}}\}_{i=1}^r$ be the principal r left*

singular vectors of Y*, and* $\{z_{\mathsf{bsh,i}}\}_{i=1}^{r}$ *be the BeamShrink weights given by* (5.13)*. As* $m, n \to \infty$*,* $m/n \to c$*, for* $i = 1, \ldots, r$*,*

$$z_{\mathsf{bsh,i}} \xrightarrow{\text{a.s.}} \sqrt{\frac{-2\,\varphi_{\mu_Z}(\rho_i)\,D'_{\mu_Z}(\rho_i)}{1 + \frac{1}{D_{\mu_Z}(\rho_i)}\left(1 - \frac{-2\,\varphi_{\mu_Z}(\rho_i) D'_{\mu_Z}(\rho_i)}{D_{\mu_Z}(\rho_i)}\right)}} \quad \text{if } \sigma_i^2 > 1/D_{\mu_Z}(b^+) \tag{5.20}$$

where $\rho_i = D^{-1}_{\mu_Z}(1/\sigma_i^2)$*. Here* $\varphi_{\mu_Z}(\cdot)$*,* $D_{\mu_Z}(\cdot)$ *and* $D'_{\mu_Z}(\cdot)$ *are transformations of* μ_Z*, defined as*

$$\varphi_{\mu_Z}(x) := \int \frac{x}{x^2 - t^2} d\mu_Z(t),$$

$$D_{\mu_Z}(x) := \left[\int \frac{x}{x^2 - t^2} d\mu_Z(t)\right] \cdot \left[c \int \frac{x}{x^2 - t^2} d\mu_Z(t) + \frac{1 - c}{x}\right], \quad x \notin \operatorname{supp}(\mu_Z),$$

$D'_{\mu_Z}(\cdot)$ *is the derivative of* $D_{\mu_Z}(\cdot)$*,* $D^{-1}_{\mu_Z}(\cdot)$ *is its functional inverse, and* $D_{\mu_z}(b^+) = \lim_{x \downarrow b} D_{\mu_Z}(x)$*.*

Proof. In [14, Theorem 2.10c)], it was shown that if $\sigma_i^2 > 1/D_{\mu_Z}(b^+)$, then for $\{i, j\} \in \{1, \ldots, r\}$, $i \neq j$, $\widehat{u}^H_{\mathsf{svd,i}} u_j \xrightarrow{\text{a.s.}} 0$. Therefore, in (5.17),

$$K_{ii} = \sum_{j=1}^{r} \frac{\sigma_j^2}{1 + \sigma_j^2}\, \widehat{u}^H_{\mathsf{svd,i}}\, u_j\, u_j^H\, \widehat{u}_{\mathsf{svd,i}} \xrightarrow{\text{a.s.}} \frac{\sigma_i^2}{1 + \sigma_i^2}\, |\widehat{u}^H_{\mathsf{svd,i}} u_i|^2 \tag{5.21}$$

Furthermore, in [14, Theorem 2.10c)], it was shown that

$$|\widehat{u}^H_{\mathsf{svd,i}} u_i|^2 \xrightarrow{\text{a.s.}} \frac{-2\,\varphi_{\mu_Z}(\rho_i)\, D_{\mu_Z}(\rho_i)}{D'_{\mu_Z}(\rho_i)} \tag{5.22}$$

where $\rho_i = D^{-1}_{\mu_Z}(1/\sigma_i^2)$, so that, by definition, $\sigma_i^2 = 1/D_{\mu_Z}(\rho_i)$. Using (5.22) in (5.21) gives us

$$K_{ii} \xrightarrow{\text{a.s.}} \frac{1}{1 + D_{\mu_Z}(\rho_i)} \cdot \frac{-2\,\varphi_{\mu_Z}(\rho_i)\, D_{\mu_Z}(\rho_i)}{D'_{\mu_Z}(\rho_i)}. \tag{5.23}$$

Substituting this limit of K_{ii} in (5.13) gives us the required result. □

V.4 Data Driven Algorithm to Estimate BeamShrink Weights

The asymptotic limit of the BeamShrink weight z_{bsh} can be computed analytically for some distributions of noise matrix Z, *e.g.*, when Z i.i.d. elements with 0 mean and variance $1/n$ [81, 14]. However, we desire a data driven algorithm using which we can estimate z_{bsh} for more general classes of distributions of Z. To that end, we define the following functions.

Definition V.1. Let $X \in \mathbb{R}_+^{m\times n}$, and $\sigma \in \mathbb{R}_+$ such that $\sigma > \max_i(X_{ii})$. We define

$$\varphi(\sigma, X) := \frac{1}{m}\operatorname{Tr}\left(\sigma(\sigma^2 I - XX^H)^{-1}\right), \tag{5.24a}$$

$$\widetilde{\varphi}(\sigma, X) := \frac{1}{n}\operatorname{Tr}\left(\sigma(\sigma^2 I - X^H X)^{-1}\right), \tag{5.24b}$$

$$D(\sigma, X) := \frac{1}{m}\operatorname{Tr}\left(\sigma(\sigma^2 I - XX^H)^{-1}\right)\cdot\frac{1}{n}\operatorname{Tr}\left(\sigma(\sigma^2 I - X^H X)^{-1}\right), \tag{5.24c}$$

$$\begin{aligned} D'(\sigma, X) := {} & \frac{1}{m}\operatorname{Tr}\left(\sigma(\sigma^2 I - XX^H)^{-1}\right)\cdot\frac{1}{n}\operatorname{Tr}\left[-2\sigma^2(\sigma^2 - X^H X)^{-2} + (\sigma^2 - X^H X)^{-1}\right] \\ & + \frac{1}{n}\operatorname{Tr}\left(\sigma(\sigma^2 I - X^H X)^{-1}\right)\cdot\frac{1}{m}\operatorname{Tr}\left[-2\sigma^2(\sigma^2 - XX^H)^{-2} + (\sigma^2 - XX^H)^{-1}\right], \end{aligned} \tag{5.24d}$$

where $\operatorname{Tr}(\cdot)$ is the trace operation.

Theorem V.3. *Suppose Y is observed data as in (5.2), and let*

$$Y = \begin{bmatrix}\widehat{U}_s & \widehat{U}_n\end{bmatrix}\begin{bmatrix}\widehat{\Sigma}_s & \mathbf{0} \\ \mathbf{0} & \widehat{\Sigma}_n\end{bmatrix}\begin{bmatrix}\widehat{V}_s & \widehat{V}_n\end{bmatrix}^H$$

be is its (ordered) SVD, where

$$\widehat{U}_s = \begin{bmatrix}\widehat{u}_{\mathsf{svd},1} & \dots & \widehat{u}_{\mathsf{svd},\mathsf{r}}\end{bmatrix}, \quad \widehat{\Sigma}_s = \operatorname{diag}(\widehat{\sigma}_{\mathsf{svd},1}, \dots, \widehat{\sigma}_{\mathsf{svd},\mathsf{r}}), \quad \widehat{V}_s = \begin{bmatrix}\widehat{v}_{\mathsf{svd},1} & \dots & \widehat{v}_{\mathsf{svd},\mathsf{r}}\end{bmatrix}.$$

Let

$$\widehat{z}_{\mathsf{bsh},\mathsf{i}} = \sqrt{\frac{-2\,\varphi(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)\,D'(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)}{1 + \dfrac{1}{D(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)}\left(1 - \dfrac{-2\,\varphi(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)\,D'(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)}{D(\widehat{\sigma}_{\mathsf{svd},\mathsf{i}}, \widehat{\Sigma}_n)}\right)}} \tag{5.25}$$

where $\varphi(\widehat{\sigma}_{\mathsf{svd}}, \widehat{\Sigma}_n)$, $D(\widehat{\sigma}_{\mathsf{svd}}, \widehat{\Sigma}_n)$, *and* $D'(\widehat{\sigma}_{\mathsf{svd}}, \widehat{\Sigma}_n)$ *were defined in* (5.24a), (5.24c) *and* (5.24d) *respectively. Then, as* $m, n \to \infty$, $m/n \to c_{\mathsf{svd}} \in (0, \infty)$, *for* $i = 1, \ldots, r$,

$$\widehat{z}_{\mathsf{bsh,i}} \xrightarrow{\text{a.s.}} z_{\mathsf{bsh,i}}. \tag{5.26}$$

Proof. The proof mirrors that in [15]. By construction, and definition of the D and φ transforms, for σ outside the support of μ_Z,

$$\varphi(\sigma, Z) \xrightarrow{\text{a.s.}} \varphi_{\mu_Z}(\sigma), \quad D(\sigma, Z) \xrightarrow{\text{a.s.}} D_{\mu_Z}(\sigma), \quad D'(\sigma, Z) \xrightarrow{\text{a.s.}} D'_{\mu_Z}(\sigma).$$

We need to show that we can estimate the spectrum of Z from the spectrum of Y. To that end, fix $r > 0$. Let $\widehat{\sigma}_{\mathsf{svd},1}, \ldots, \widehat{\sigma}_{\mathsf{svd,r}}$ denote the principal r singular values of Y ($\widehat{\sigma}_{\mathsf{svd}} = \widehat{\sigma}_{\mathsf{svd},1}$ in this notation), and let

$$\widehat{\Sigma}_n = \operatorname{diag}(\widehat{\sigma}_{\mathsf{svd,r+1}}, \ldots, \widehat{\sigma}_{\mathsf{svd,m}}).$$

Define

$$\mu_{Y,r} = \frac{1}{m-r} \sum_{i=r+1}^{m} \delta_{\widehat{\sigma}_{\mathsf{svd,i}}}.$$

Then, the proof in [15, Section 2.5] shows that, for fixed r, if

$$\varphi(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n) \xrightarrow{\text{a.s.}} \varphi_{\mu_Z}(\rho_i), \quad D(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n) \xrightarrow{\text{a.s.}} D_{\mu_Z}(\rho_i), \quad D'(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n) \xrightarrow{\text{a.s.}} D'_{\mu_Z}(\rho_i), \tag{5.27}$$

if $\widehat{\sigma}_{\mathsf{svd,i}} \to \rho_i > b$. Using the convergences of the quantities in (5.27) in (5.25) completes the proof. □

Theorem V.3 tells us that $\{\widehat{z}_{\mathsf{bsh,i}}\}_{i=1}^r$ in (5.25) are consistent estimates of $\{z_{\mathsf{bsh,i}}\}_{i=1}^r$ in (5.13), and thus gives us a data driven way to compute the BeamShrink weight. Armed with these estimates, we propose Algorithm 5.1, which returns data driven BeamShrink weights.

Given BeamShrink weights output from Algorithm 5.1, we can form the BeamShrink based beamformer

$$\widehat{w}_{\mathsf{bsh}} := \frac{\widehat{R}_{\mathsf{bsh}}^{-1} u_t}{\|\widehat{R}_{\mathsf{bsh}}^{-1} u_t\|_2}, \tag{5.28}$$

where

$$\widehat{R}_{\mathsf{bsh}} := I + \widehat{U}_s \operatorname{diag}(\widehat{z}_{\mathsf{bsh},1}^2, \ldots, \widehat{z}_{\mathsf{bsh,r}}^2) \widehat{U}_s^H. \tag{5.29}$$

Algorithm 5.1: BeamShrink

Input: $Y \in \mathbb{C}^{m \times n}$, data matrix

Input: r, rank

Output: $\widehat{z}_{\mathsf{bsh}} = \{z_{\mathsf{bsh,i}}\}_{i=1}^{r}$

Compute $Y = \widehat{U}\widehat{\Sigma}\widehat{V}^H$, the SVD of Y

Let $\widehat{\Sigma}_s = \widehat{\Sigma}(1:r, 1:r), \quad \widehat{\Sigma}_n = \widehat{\Sigma}(r+1:m-1, r+1:n-1)$

for $i = 1$ **to** r **do**

  Let $\widehat{\sigma}_{\mathsf{svd,i}} = \widehat{\Sigma}_s(i,i)$

  Compute $\varphi(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n)$, $D(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n)$, $D'(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_n)$ from (5.24a), (5.24c), (5.24d)

  Compute K_{ii} from (5.23)

  Compute $\widehat{z}_{\mathsf{bsh,i}} = \sqrt{\dfrac{1 - K_{ii}}{K_{ii}}}$

end

Sort $\{\widehat{z}_{\mathsf{bsh,i}}\}$ in descending order

V.5 Numerical Simulations

We apply BeamShrink to STAP. The STAP problem was outlined in Section III.7. Recall from 3.50 that the array manifold vector in this case is parameterized by two parameters: angle and Doppler frequency. The manifold vector has the form

$$u := u(\theta, f)$$

where θ is the angle parameter and f is the Doppler frequency parameter. Recall from (3.54) that the notch depth at clutter location corresponding to steering vector $u_{c,i}$ of a given beamformer w is defined as

$$|w^H u_{c,i}|^2.$$

The beampattern of a given vector w is defined as

$$\mathrm{BP}(\theta, f) := |w^H u(\theta, f)|^2. \tag{5.30}$$

Given an estimate of the covariance matrix $\widehat{R}$, the normalized signal-to-interference noise ratio (SINR) is defined as

$$\mathrm{SINR}(\theta, f) := \frac{|u(\theta, f)^H \widehat{R}^{-1} u(\theta, f)|^2}{\left(u(\theta, f)^H \widehat{R}^{-1} R \widehat{R}^{-1} u(\theta, f)\right) \left(u(\theta, f)^H R^{-1} u(\theta, f)\right)}, \tag{5.31}$$

where R is the population covariance matrix.

We observe data Y as in (5.2). From this, we compute the SVD based beamformer as in (5.10), in which the SVD based covariance matrix is computed as in (5.8). We also compute BeamShrink weights using Algorithm 5.1, and thus form the BeamShrink based beamformer (5.28). We refer to the weights obtained from solving the optimization problem (5.12) numerically as "exact solution".

Rank One Signal

Fig. 5.1 plots the average notch depth (over 1000 trials) of the beamformer for a STAP system with a single clutter source (*i.e.*, rank one signal), for $m = 300$, $n = 250$. The clutter is located at 30^o and a target at 45^o. Array elements are spaced half wavelength apart, and the pulse repetition frequency is 300 Hz. We see that BeamShrink based beamformer gives deeper notch depth relative to the SVD based beamformer at lower interferer-to-noise ratio (*i.e.*, $\sigma_{c,1}$).

Fig. 5.2 plots the average log-error $:= \log(\|R^{-1} - \widehat{R}_z^{-1}\|_F^2)$, for $z = \widehat{z}_{\mathsf{svd}}$ as in (5.9), and $z = \widehat{z}_{\mathsf{bsh}}$. We also plot the error due to the exact solution for comparison. We see that BeamShrink weights have lower error at lower σ_1, which accounts for the improved performance in Fig. 5.1.

Fig. 5.3 plots the average weights (over 1000 trials) returned by the SVD and BeamShrink algorithms vs $\sigma_{c,1}$, for the system in Fig. 5.1. The plot illustrates the nature of the shrinkage. The BeamShrink weights are equal to the exact solution, as suggested by Theorem V.2.

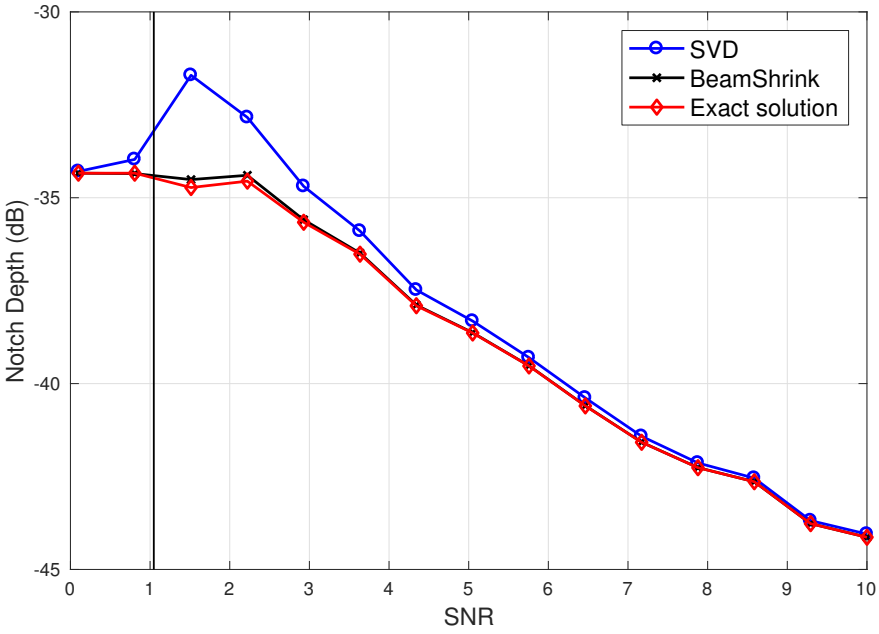


Figure 5.1: Plot of average notch depth of SVD based beamformer (5.10), BeamShrink based beamformer (5.28), and beamformer based on exact solution vs SNR σ_1, for a system with $m = 300$, $n = 250$. We see that the BeamShrink based beamformer has a deeper notch depth at lower values of $\sigma_{c,1}$.

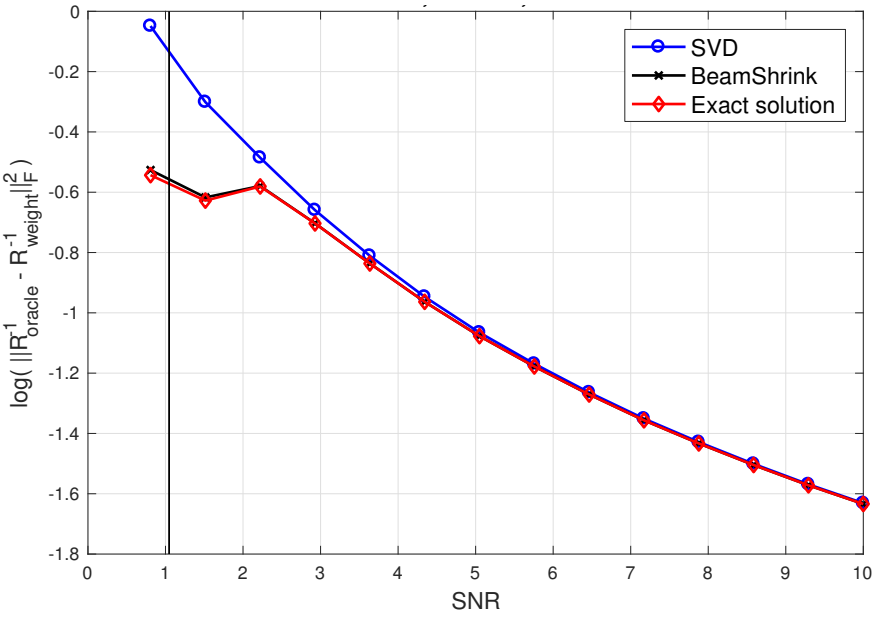


Figure 5.2: Plot of average error $\|R^{-1} - \widehat{R}_z^{-1}\|_F^2$ when $\widehat{R}_z$ uses SVD weights and BeamShrink weights. At low $\sigma_{c,1}$, BeamShrink weights yield smaller error than SVD weights, leading to improved denoising. The average error of the data driven BeamShrink weight is equal to the that achieved by using the exact solution, as suggested by Theorem (V.2).

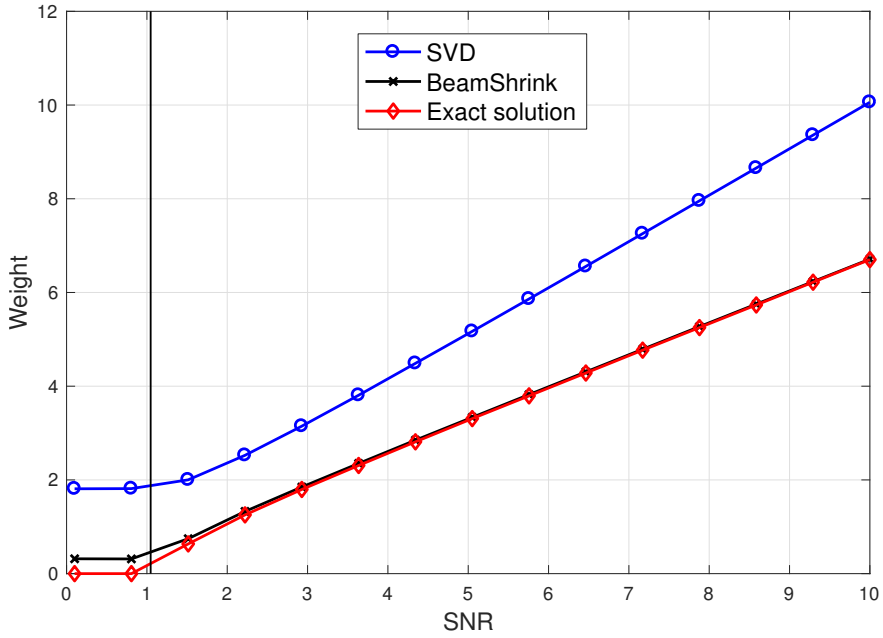
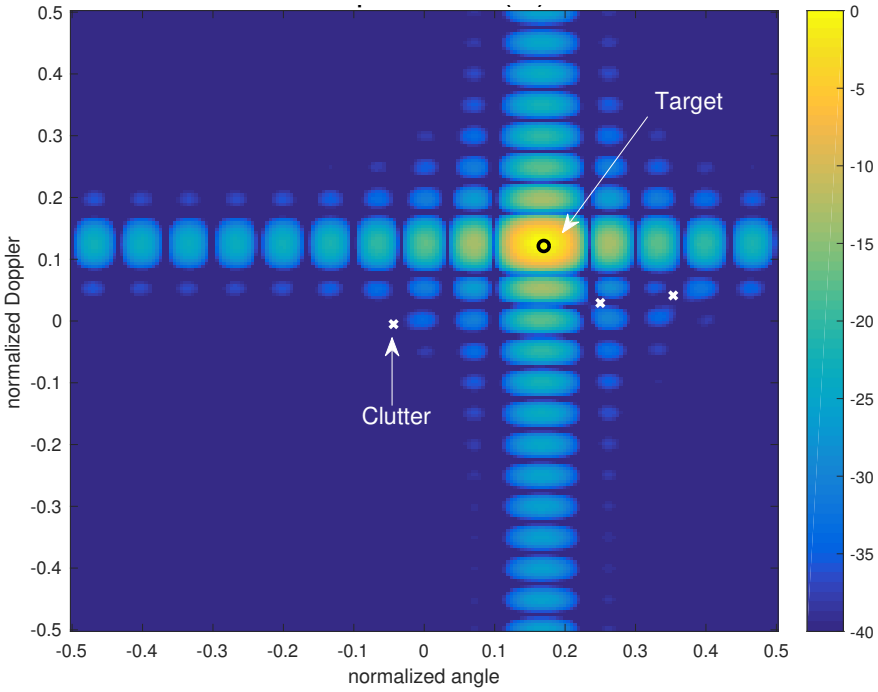


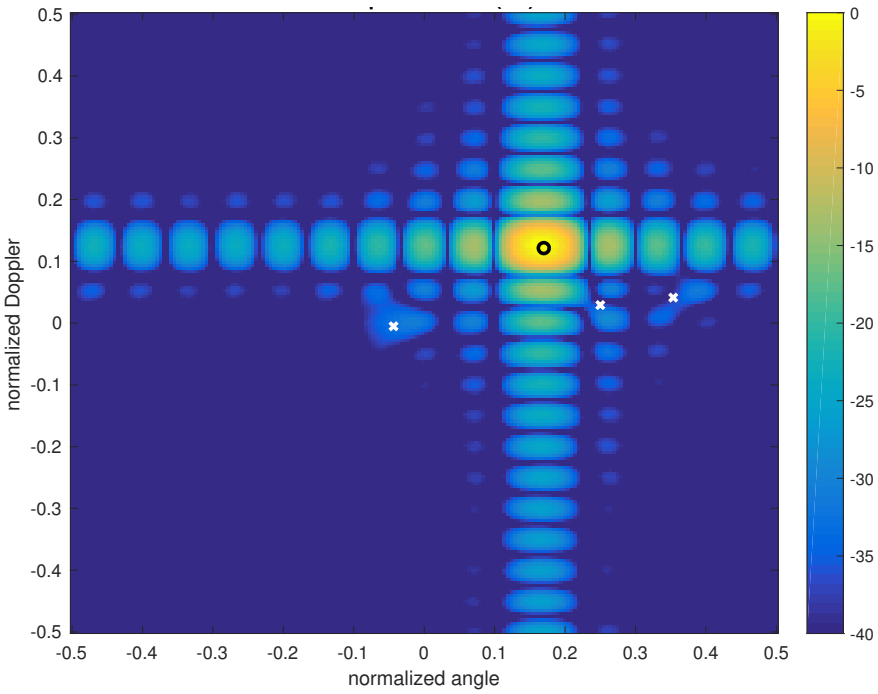
Figure 5.3: Plot of SVD and BeamShrink weights as a function of σ_1, for the system in Fig. 5.1. Here we see that the optimum weight for denoising is a shrinkage of the SVD weight. We also see that the data driven weight $\widehat{z}_{\mathsf{bsh}}$ matches the exact solution.

Rank Three Signal, Input Rank equals True Rank

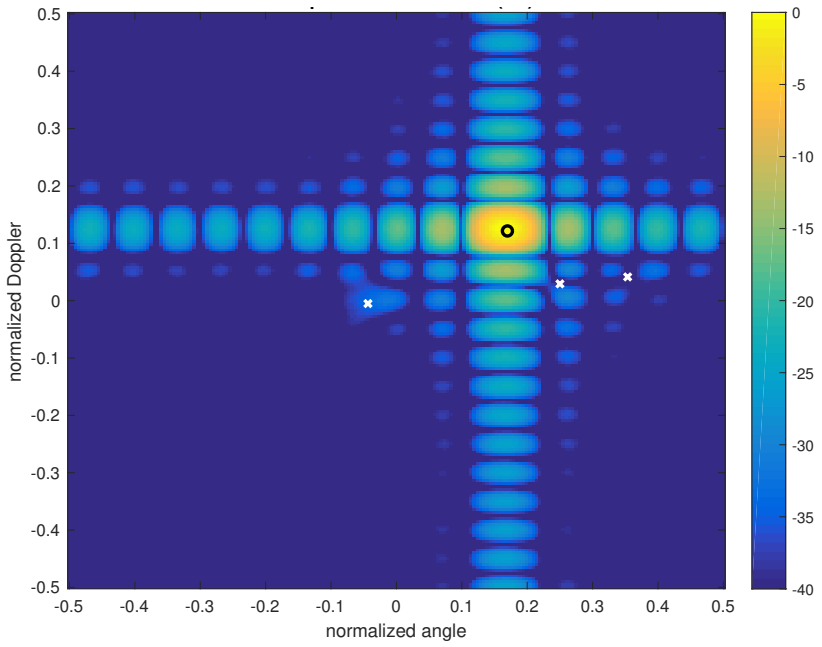
In Fig. 5.4 we plot the beampatterns for a system with 3 clutter sources (rank 3 signal), for $m = 300$, $n = 150$. Fig. 5.4a is the oracle beampattern, when the beamformer is computed using the population covariance matrix. Fig. 5.4b is the beampattern due to the SVD based beamformer, and Fig. 5.4c is due to the BeamShrink based beamformer. In each case, the input rank – the guess for number of clutter sources – was also 3, so that there was no rank mismatch while picking the top r principal components of the SVD to estimate the clutter covariance matrix. There is no obvious difference in the beampatterns of the SVD beamformer and BeamShrink beamformer.



(a) Oracle beampattern computed using (5.30) of a system with $m = 300$, $n = 150$. The beampattern is a measure of directional energy of the beamformer. It is desirable to have peak power at target location, while suppressing power output at clutter locations. The clutter locations are marked with a white 'x', while the target location is marked with a black circle.



(b) Beampattern of the SVD based beamformer, for the system in Fig. 5.4a. The rank of clutter subspace, r, is known, and the beamformer is computed using r principal components of the SVD of Y.

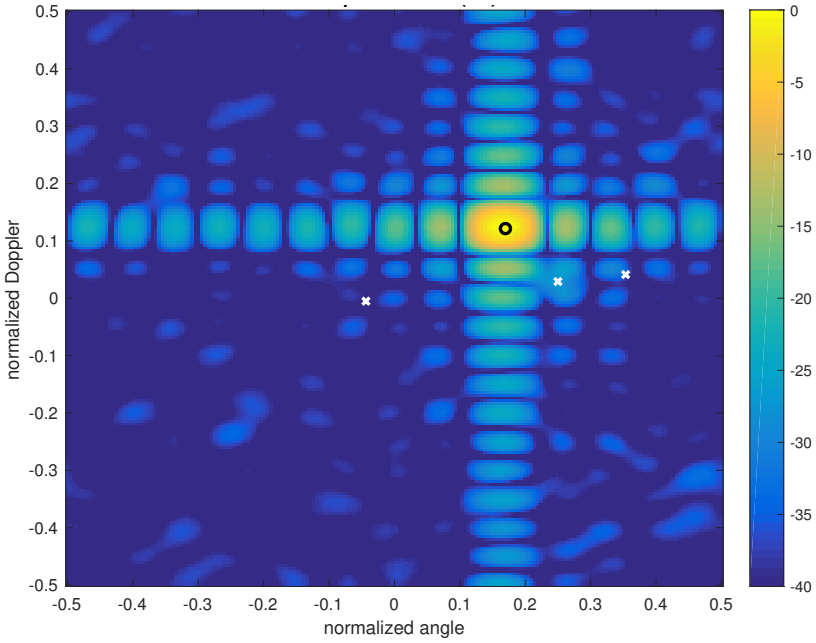


(c) Beampattern of the BeamShrink based beamformer for the system in Fig. 5.4. The rank of clutter subspace, r, is known, and r BeamShrink weights are computed from Algorithm 5.1.

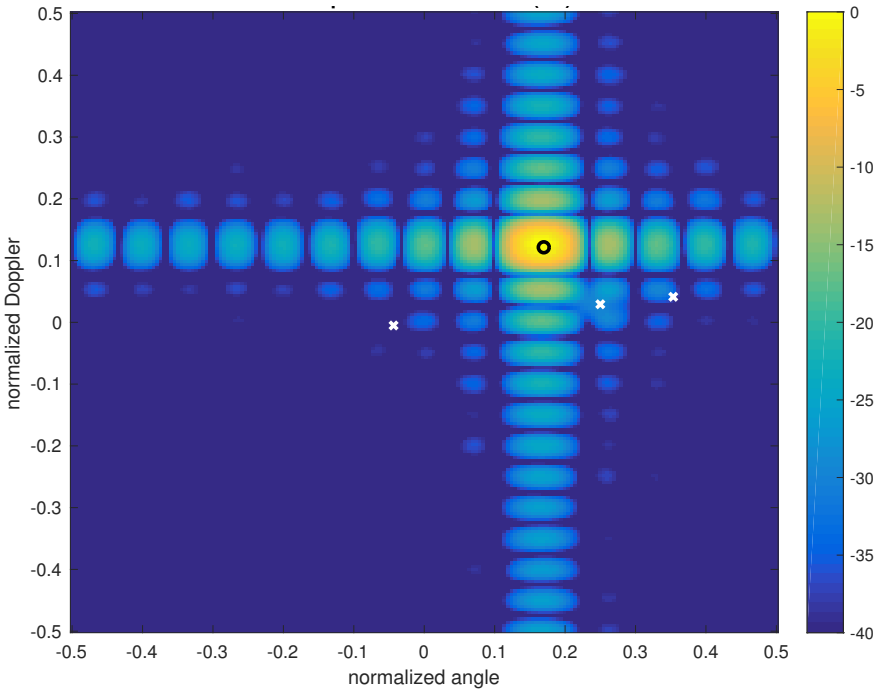
Figure 5.4: Beampatterns of oracle, SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$, and 3 sources of clutter.

Rank Three Signal, Input Rank Eight

In Fig. 5.4 we plot the beampatterns for the same system as in Fig. 5.4, but we overestimate the rank of the clutter subspace. By input rank eight, we mean that we use the top eight principal components of the SVD to estimate the covariance matrix (5.8), and compute up to eight BeamShrink weights in Algorithm 5.1. Since BeamShrink shrinks uninformative components in a data driven way, the problem of rank overestimation is mitigated.



(d) Beampattern of the SVD beamformer for the system in Fig. 5.4, but now the rank of the clutter subspace has been overestimated. We use the top 7 principal components of the SVD of Y to compute the SVD based beamformer. As a result, we see a clear deviation in the beampattern from the oracle beampattern of Fig. 5.4a.



(e) Beampattern of the BeamShrink based beamformer for the system in Fig. 5.4a. We assume that the rank of the clutter subspace has been overestimated, and so we use the top 7 singular vectors of Y and 7 corresponding BeamShrink weights to compute the beamformer. In spite of the rank overestimation, we see that the Beampattern has not deviated much from the plot in Fig. 5.4c. This is because the uninformative components corresponding to the higher rank components are shrunk down by BeamShrink.

Figure 5.4: Beampatterns of the SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$ and 3 clutter sources, when the input rank is 7.

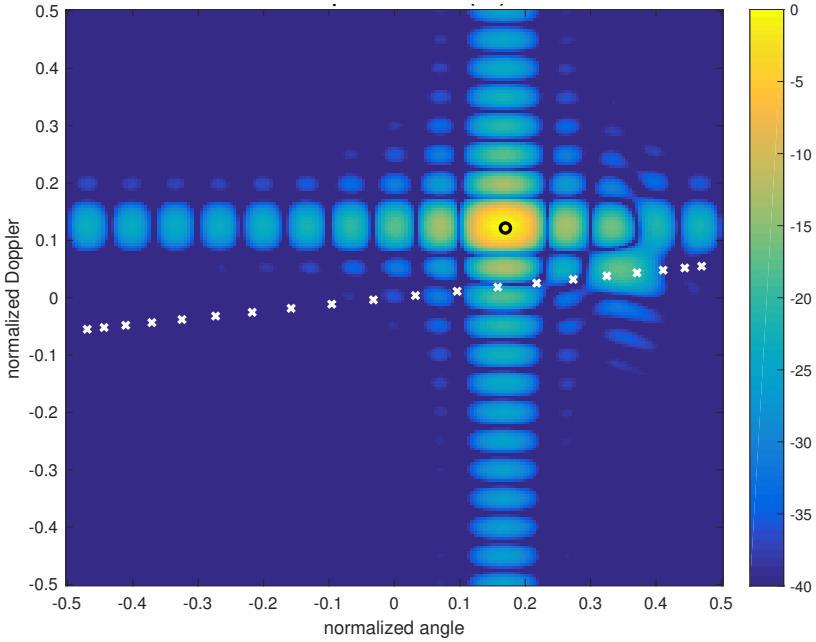
Rank 20 Signal, Effective Rank 12

In Fig. 5.5 we plot the beampatterns for a system with 20 clutter sources, for a system with $m = 300$, $n = 150$. By effective rank, we mean

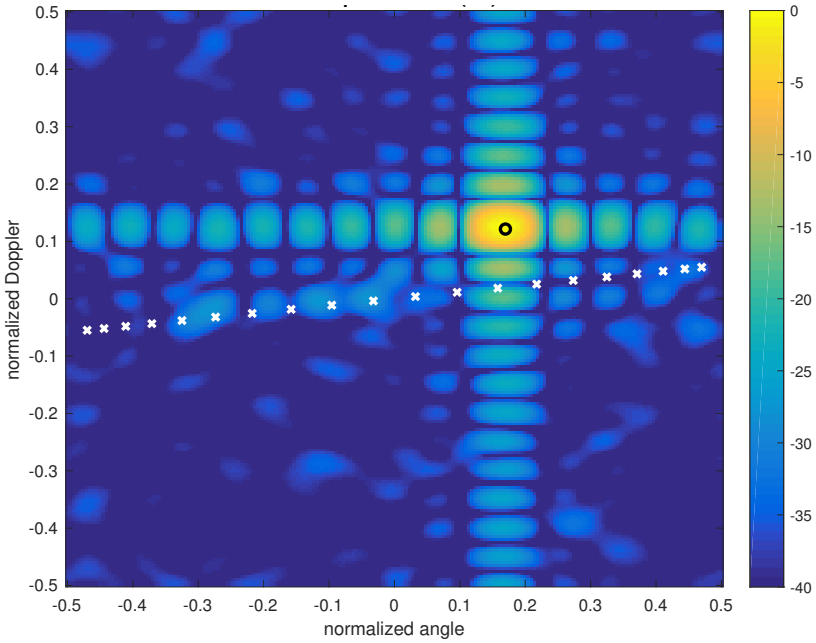
$$\text{number of } \sigma_i > 1/D_{\mu_Z}(b^+).$$

Here, even though we are not overestimating the input rank, the number of informative signal components is lower than 20. Therefore, we see that BeamShrink does better than the SVD in a manner similar to Fig. 5.4.

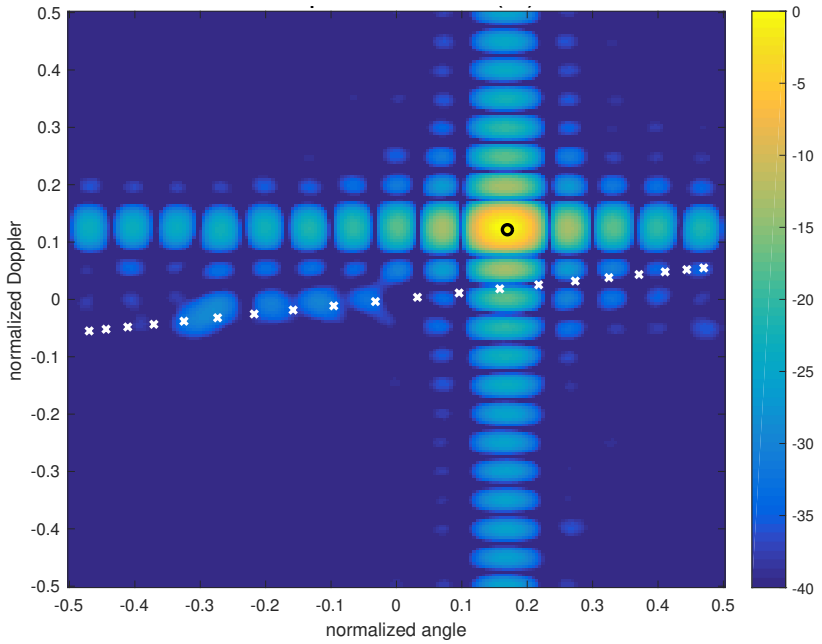
In Fig. 5.6, we plot the average SINR (over 100 trials), normalized relative to the true covariance matrix, for the SVD beamformer and the BeamShrink beamformer. From Fig. 5.6c, we see that the BeamShrink beamformer gives us an SINR gain of 0.14 dB at target location.



(a) Oracle beampattern for a system with $m = 300$, $n = 150$, and 20 clutter sources.

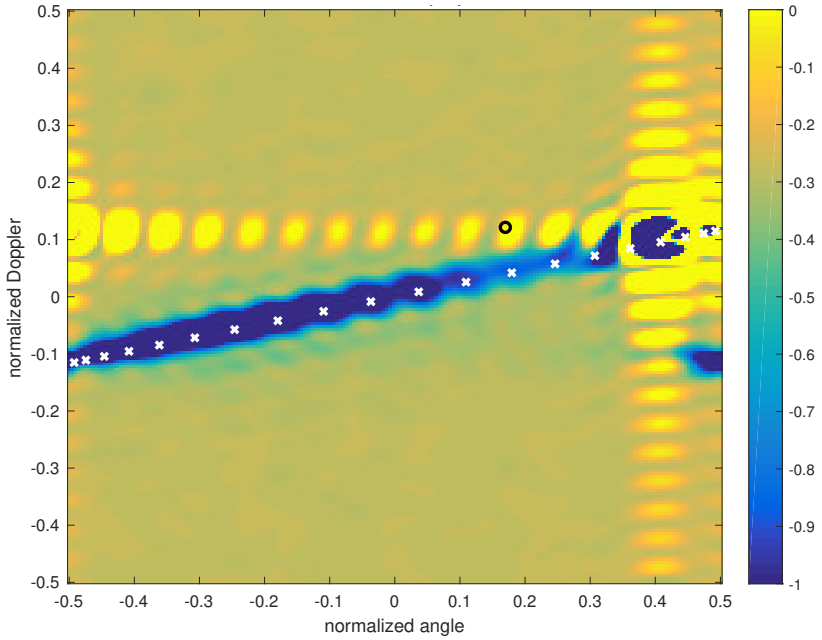


(b) Beampattern of the SVD based beamformer for the system in Fig. 5.5a. The input rank to compute the beamformer is 20, which is exactly the number of clutter sources. However, since the SNR of some of the sources is below the critical SNR, the effective rank of the clutter subspace is 12. The addition of uninformative components in the computation of the beamformer results in several high sidelobes in the beampattern.

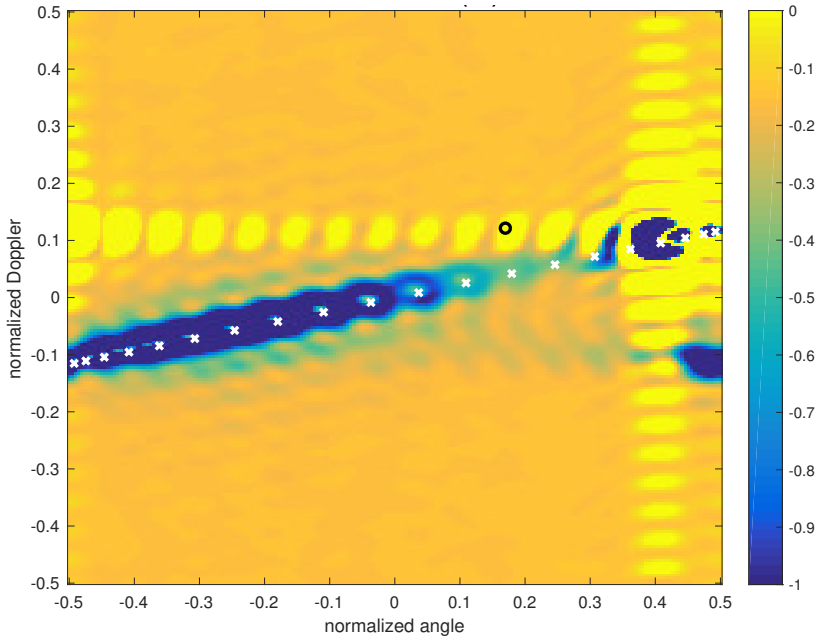


(c) Beampattern of the BeamShrink based beamformer for the system in Fig. 5.5a. The input rank to compute the beamformer is 20, which is exactly the number of clutter sources, but the effective rank of the system is 12. Despite the mismatch between the input rank and number of informative components, the performance of the BeamShrink based beamformer does not degrade as much as it does for the SVD based beamformer, since the uninformative components are shrunk down.

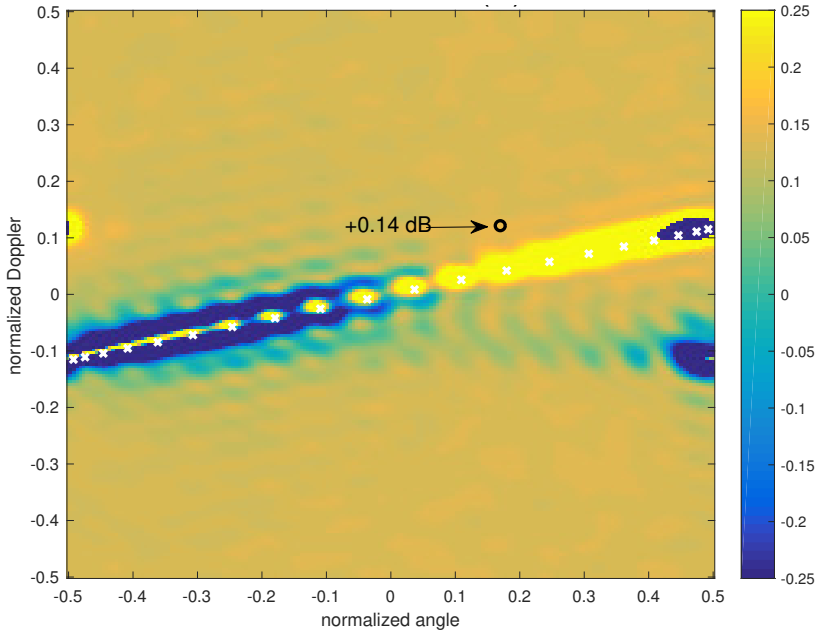
Figure 5.5: Beampattern for SVD and BeamShrink based beamformers for a system with $m = 300$, $n = 150$, 20 clutter sources of which 12 are informative.



(a) Normalized SINR of the SVD based beamformer, averaged over 100 trials, for the system in Fig. 5.5b. An ideal beamformer would return 0 throughout the heatmap. A higher SINR is desirable since it results in easier target detection amidst clutter.



(b) Normalized SINR of the BeamShrink based beamformer, averaged over 100 trials, for the system in Fig. 5.5b. We see that the BeamShrink beamformer returns higher SINR values at a majority of (angle, Doppler) locations. Crucially, it results in a higher SINR at target and clutter locations.



(c) Difference of BeamShrink beamformer SINR and SVD beamformer SINR of Figs. 5.6a and 5.6b. The BeamShrink based beamformer provides an SINR gain of $0.14dB$ at target location.

Figure 5.6: Normalized SINR of SVD based beamformer and BeamShrink beamformers for a system with $m = 300$, $n = 150$, 20 clutter sources, of which 12 are informative.

Rank 20 Signal, Effective Rank 12, Input Rank 25

We maintain the same setup from Fig. 5.6, in which the number of informative components was lesser than the rank of the signal, but we also overestimate the input rank to 25. In Fig. 5.7, we plot the average (normalized) SINR difference (over 100 trials), between the BeamShrink and SVD beamformers. We see that the BeamShrink beamformer gives us an SINR gain of 0.14 dB at target location.

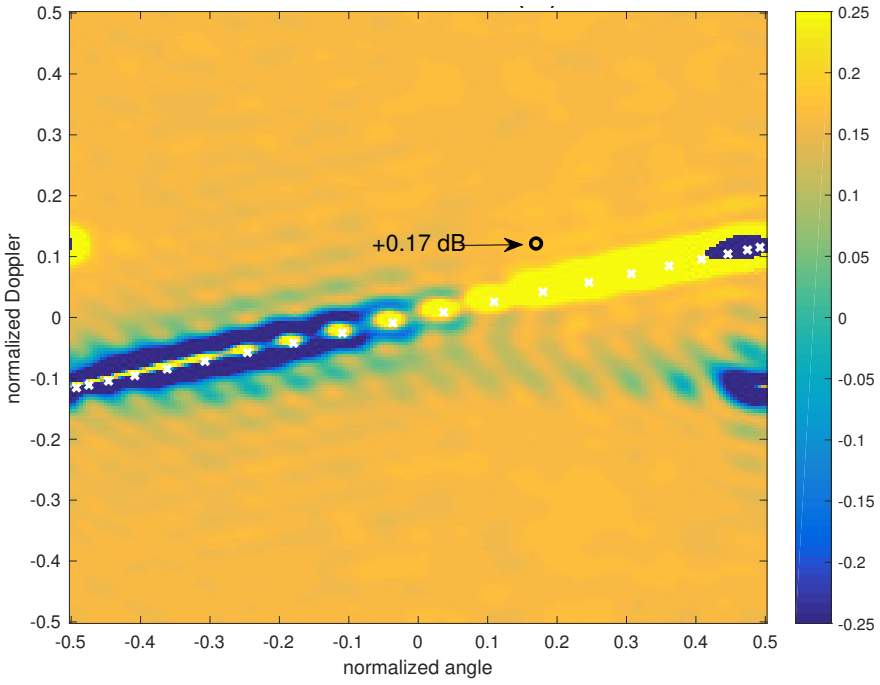


Figure 5.7: Difference between SINRs due to BeamShrink and SVD based beamformers for the system in Fig. 5.6. In contrast with Fig. 5.6, the input rank is 25, which is an overestimation of the number of clutter sources. There is a rank mismatch both due to input rank and number of informative components. Since BeamShrink is robust to rank overestimation compared to the SVD, we see that the BeamShrink beamformer yields a higher SINR gain at target: 0.17 dB compared to 0.14 dB in Fig. 5.6c.

V.6 Conclusions

In this chapter, we considered the problem of improved inverse covariance matrix estimation, for the application to SVD based beamformers. We derived showed that the estimation can be improved by re-weighting the sample singular vectors. We derived exact optimum re-weighting for the rank one case, and an approximate re-weighting for the higher rank case.

We showed that these weights are a non-convex shrinkage operation on the corresponding underlying singular value. We proposed a data driven algorithm called BeamShrink for the estimation of these weights. We showed that BeamShrink led to improved beamformer performance using numerical simulations.

CHAPTER VI

TensorShrink: An Algorithm for Improved Denoising of Low Rank Tensor Estimates

VI.1 Introduction

Tensor valued signals have wide ranging applications, such as in independent component analysis [82], blind source separation [83, 84], bioinformatics [85] and latent variable models [86], and so on.

There has been much recent work in the field of low rank tensor estimation [87, 88, 63]. There are two widely used models for low rank tensor decomposition in the literature. The first is the Tucker decomposition [63, 64]. The Tucker decomposition/HOSVD approach to the decomposition takes as its input a dense tensor of order k, and returns k orthonormal matrices and a dense core tensor. The input matrix is recovered by multiplying the orthogonal matrices with the core tensor at the corresponding order/face of the tensor. Many algorithms proposed for Tucker decomposition are finite (non-iterative) [59, 62, 60, 89, 61], and involve multiple SVDs of tensors unfolded into matrices. Other approaches include gradient descent based algorithms [90, 91], convex optimization [88, 92] etc.

The CANDECOMP/Parafac (CP) decomposition returns k non-orthogonal tensors, and a superdiagonal core tensor [93, 94]. It is generally computed using alternating least squares (ALS). Mature numerical packages and toolboxes are available with standard algorithms for low rank tensor estimation [95, 96].

In this chapter, we are concerned with the estimation of low rank tensors. Let $\mathcal{Y} = \mathcal{S} + \mathcal{Z}$, where observed tensor $\mathcal{Y}$ is the sum of signal $\mathcal{S}$ and noise $\mathcal{Z}$. Tensor algorithms in the

literature minimize the representation error:

$$\|\mathcal{Y} - \widehat{\mathcal{S}}\|_F^2,$$

where $\mathcal{Y}$ is the observed data tensor and $\widehat{\mathcal{S}}$ is a low rank estimate. However, in many signal processing applications, the quantity that we wish to minimize is the denoising error:

$$\|\mathcal{S} - \widehat{\mathcal{S}}\|_F^2.$$

The problem of minimizing denoising error is an unobservable problem, as $\mathcal{S}$ is not known. However, we can use the approach and philosophy of [15] and Chapter V of this thesis. We can draw upon any tensor estimation algorithm, such as the ALS algorithm for CP decomposition, and re-weight the terms in output tensor decomposition to achieve denoising.

The bottleneck to completing the solution via this approach is that the asymptotic quantification of the terms involved in the expression for optimal weight are open problems. As we will show in this chapter, at the heart of finding optimal weights for denoising are terms which account for the accuracy of estimation of the factors in the tensor decomposition with respect to the true underlying factors. The thrust of this chapter is to find mathematically justified approximations to these terms, and propose a suboptimal, data driven algorithm which nonetheless provides improved denoising for low rank tensor estimation problems. We terms these weights "TensorShrink".

We also combine the ideas of TensorShrink with those of Chapter V, and propose an algorithm called TensorBeamshrink, which leads to the improved performance of beamformers computed using a low rank tensor approximation based approach.

Due to the nature of approximations involved, we will see that the resulting weights for denoising are shrinkage weights, which penalize any factor which is not an accurate estimate.

The rest of this chapter is organized as follows. We introduce the model and low rank estimation problem in Section VI.2. We present the ALS algorithm, and make some notes on its initialization in Section VI.3. We derive expresions for TensorShrink weights for rank one and higher rank signals in Section VI.4. We present an algorithm that returns data driven TensorShrink weights in Section VI.5. We present the TensorBeamshrink algorithm in Section VI.6. We validate our analysis via numerical simulations in Section VI.7, and present concluding remarks in Section VI.8.

VI.2 Data Model and Low Rank Estimation

Suppose we observe the tensor $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, which has the form

$$\mathcal{Y} = \underbrace{\sum_{i=1}^{r} \sigma_i \, \bar{s}_i \circ b_i \circ a_i}_{:=\mathcal{S}} + \mathcal{Z}, \tag{6.1}$$

where $\{\sigma_i\}_{i=1}^r \in \mathbb{R}_+$, $\{a_i\}_{i=1}^r \in \mathbb{C}^{m_1}$, $\{b_i\}_{i=1}^r \in \mathbb{C}^{m_2}$, $\{s_i\}_{i=1}^r \in \mathbb{C}^n$ are unit norm vectors, $\circ$ denotes outerproduct, and $\mathcal{Z} \in \mathbb{C}^{n \times m_1 \times m_2}$ is a noise only tensor whose elements are mean 0, variance $1/n$, and have bounded higher order moments. Here $\bar{s}_i$ indicates complex conjugate of s_i. The motivation behind choosing $\bar{s}_i$ instead of s_i is to keep the notation of (6.1) consistent with (3.3). In accordance with usual notation, we denote the $(i,j,k)^{\text{th}}$ entry of $\mathcal{Y}$ as $\mathcal{Y}_{ijk}$.

The low rank decomposition of $\mathcal{Y}$ has the form

$$\widehat{\mathcal{Y}} = \sum_{i=1}^{r} \widehat{\sigma}_i \, \widehat{\bar{s}}_i \circ \widehat{b}_i \circ \widehat{a}_i, \tag{6.2}$$

where $\{\widehat{\sigma}_i\} \in \mathbb{R}_+$, and $\{\widehat{a}_i, \widehat{b}_i, \widehat{\bar{s}}_i\}$ are unit norm vectors which are estimates of the corresponding underlying quantities. Without loss of generality, we assume that the components are ordered such that $\widehat{\sigma}_1 \geq \widehat{\sigma}_2 \geq \ldots \widehat{\sigma}_r$.

The Frobenius norm of tensor $\mathcal{Y}$ is defined as

$$\|\mathcal{Y}\|_F = \sqrt{\sum_{ijk} \mathcal{Y}_{ijk}^2}. \tag{6.3}$$

Most existing estimation algorithms in the literature aim to minimize the *representation error* in the Frobenius norm sense. Given estimates $\{\widehat{\sigma}_i, \widehat{a}_i, \widehat{b}_i, \widehat{s}_i\}$, the representation error is given by:

$$\left\| \mathcal{Y} - \sum_{i=1}^{r} \widehat{\sigma}_i \, \widehat{\bar{s}}_i \circ \widehat{b}_i \circ \widehat{a}_i \right\|_F^2. \tag{6.4}$$

In many signal processing problems,the quantity which is of real interest is the *denoising error*, which is given by:

$$\left\| \mathcal{S} - \sum_{i=1}^{r} \widehat{\sigma}_i \, \widehat{\bar{s}}_i \circ \widehat{b}_i \circ \widehat{a}_i \right\|_F^2. \tag{6.5}$$

Algorithms which minimize representation (6.4) are not necessarily optimal with respect to denoising error (6.5). Let

$$\widehat{\mathcal{S}}(z) := \sum_{i=1}^{r} z_i \, \widehat{\overline{s}}_i \circ \widehat{b}_i \circ \widehat{a}_i, \tag{6.6}$$

be the estimated tensor, with weights $\{z_i\}$ in place of $\{\widehat{\sigma}_i\}$ returned by the canonical estimation algorithm (See Section VI.3). Consider the optimization problem

$$z_{\mathsf{tsh}} = \underset{z \in \mathbb{R}_+^r}{\arg\min} \, \mathrm{SE}(z) := \underset{z \in \mathbb{R}_+^r}{\arg\min} \left\| \mathcal{S} - \widehat{\mathcal{S}}(z) \right\|_F^2. \tag{6.7}$$

By construction, the estimate $\widehat{\mathcal{S}}(z_{\mathsf{bsh}})$ will have lower denoising error than the canonical low rank estimate $\widehat{S}(\widehat{\sigma})$ $(= \widehat{\mathcal{Y}})$. We refer to the weights

$$z_{\mathsf{tsh}} := \begin{bmatrix} z_{\mathsf{tsh},1} & \dots & z_{\mathsf{tsh},\mathsf{r}} \end{bmatrix}^T$$

as "TensorShrink" weights, since, as we will see in Section VI.4, they will be a shrinkage of $\{\sigma_i\}$.

VI.3 The ALS Algorithm and Its Initialization

Before we proceed to derive z_{tsh}, we will describe the tensor estimation algorithm which we use throughout this chapter. Several algorithms exist for low rank tensor estimation. Our choice of algorithm is the popular Alternating Least Squares (ALS).

Denote by

$$\widehat{A} = \begin{bmatrix} \widehat{a}_1 & \dots & \widehat{a}_r \end{bmatrix}, \quad \widehat{B} = \begin{bmatrix} \widehat{b}_1 & \dots & \widehat{b}_r \end{bmatrix}, \quad \widehat{S} = \begin{bmatrix} \widehat{s}_1 & \dots & \widehat{s}_r \end{bmatrix}$$

the estimates of the corresponding underling quantities,

$$A = \begin{bmatrix} a_1 & \dots & a_r \end{bmatrix}, \quad B = \begin{bmatrix} b_1 & \dots & b_r \end{bmatrix}, \quad S = \begin{bmatrix} s_1 & \dots & s_r \end{bmatrix}.$$

Furthermore, let

$$U = A \circledast B, \quad \widehat{U} = \widehat{A} \circledast \widehat{B}, \tag{6.8}$$

where $\circledast$ denotes Khatri-Rao product [97, 98], and let

$$\Sigma_s = \text{diag}(\sigma_1, \ldots, \sigma_r).$$

The ALS algorithm takes as its input the tensor $\mathcal{Y}$ and rank r, and returns estimates $\widehat{A}, \widehat{B}$ as in (6.8). For completeness, we outline the ALS algorithm in 6.1.

Algorithm 6.1: ALS Algorithm for Low Rank Tensor Estimation

Input: $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, data tensor

Output: $\widehat{A} \in \mathbb{C}^{m_1 \times r}, \widehat{B} \in \mathbb{C}^{m_2 \times r}$

Compute $Y = \mathcal{Y}_{[1]}$, unfolded tensor

Compute initial estimates $\widehat{A}, \widehat{B}$

while *Not converged* **do**

  $\widehat{\overline{S}} \longleftarrow$ updated LS solution

  `// Optimization problem:` $\widehat{\overline{S}} = \arg\min_{S} \|Y_{[1]} - S\Sigma_s(\widehat{A} \circledast \widehat{B})^T\|_F^2$

  $\widehat{B} \longleftarrow$ updated LS solution

  `// Optimization problem:` $\widehat{B} = \arg\min_{B} \|Y_{[2]} - B\Sigma_s(\widehat{\overline{S}} \circledast \widehat{A})^T\|_F^2$

  $\widehat{A} \longleftarrow$ updated LS solution

  `// Optimization problem:` $\widehat{A} = \arg\min_{A} \|Y_{[3]} - A\Sigma_s(\widehat{\overline{S}} \circledast \widehat{B})^T\|_F^2$

end

Using the outputs $\{\widehat{a}_i, \widehat{b}_i\}$ from the ALS algorithm, we determine $\{\widehat{\sigma}_i, \widehat{s}_i\}$ are computed as shown in Fig. 6.1.

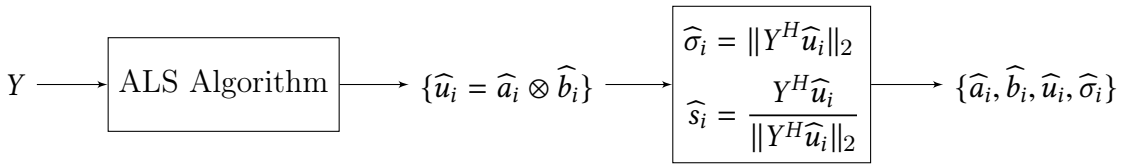


Figure 6.1: Procedure to find estimates $\{\widehat{\sigma}_i, \widehat{s}_i\}$ given $\{\widehat{a}_i, \widehat{b}_i\}$.

A complete study of the convergence of the ALS algorithm, and the effect of initialization

is beyond the scope of this work. Canonically, the ALS algorithm is initialized by reshaping the data tensor and computing the principal components of the resulting matrix, as described in Algorithm 6.2.

Algorithm 6.2: SVD Based Initialization for Tensor Estimation

Input: $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, data tensor

Output: $\widehat{A} \in \mathbb{C}^{m_1 \times r}, \widehat{B} \in \mathbb{C}^{m_2 \times r}$

1 Compute $\widehat{B}$ = principal r left singular vectors of $\mathcal{Y}_{[2]}$ // $\mathcal{Y}_{[2]} = B\Sigma_s(\widehat{\widehat{S}} \circledast \widehat{A})^T + \mathcal{Z}_{[2]}$

2 Compute $\widehat{A}$ = principal r left singular vectors of $\mathcal{Y}_{[3]}$ // $\mathcal{Y}_{[3]} = A\Sigma_s(\widehat{\widehat{S}} \circledast \widehat{B})^T + \mathcal{Z}_{[3]}$

Algorithm 6.3: Columnwise KPCA Based Initialization for Tensor Estimation

Input: $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, data tensor

Output: $\widehat{A} \in \mathbb{C}^{m_1 \times r} = [\widehat{a}_1, \ldots, \widehat{a}_r]$, $\widehat{B} \in \mathbb{C}^{m_2 \times r} = [\widehat{b}_1, \ldots, \widehat{b}_r]$

1 Compute $\widehat{U}_{\text{svd}}$ = principal r left singular vectors of $\mathcal{Y}_{[1]}^T$ // $\mathcal{Y}_{[1]}^T = (A \circledast B)\Sigma_s S^H$

2 **for** $i = 0$ **to** r **do**

3 Let $\widehat{u}_{\text{svd,i}} = \widehat{U}_{\text{svd}}(:, i)$

4 Compute $\widehat{U}_{R,i} = \mathcal{R}_{m_1 m_2 \times 1 \to m_1 \times m_2}(\widehat{u}_{\text{svd,i}})$

5 Compute $\widehat{a}_{\text{ckpca,i}}$ = principal left singular vector of $\widehat{U}_{R,i}$

6 Compute $\widehat{b}^*_{\text{ckpca,i}}$ = principal right singular vector of $\widehat{U}_{R,i}$

7 **end**

In this work, we use a multirank analogue of the KPCA2, 1 (Algorithm 3.1), which is described in Algorithm 6.3. The motivation for using columnwise KPCA2 based initialization is seen in Fig. 6.2 and Fig. 6.3: in numerical simulations, we found that using this initialization yielded faster convergence of the ALS algorithm.

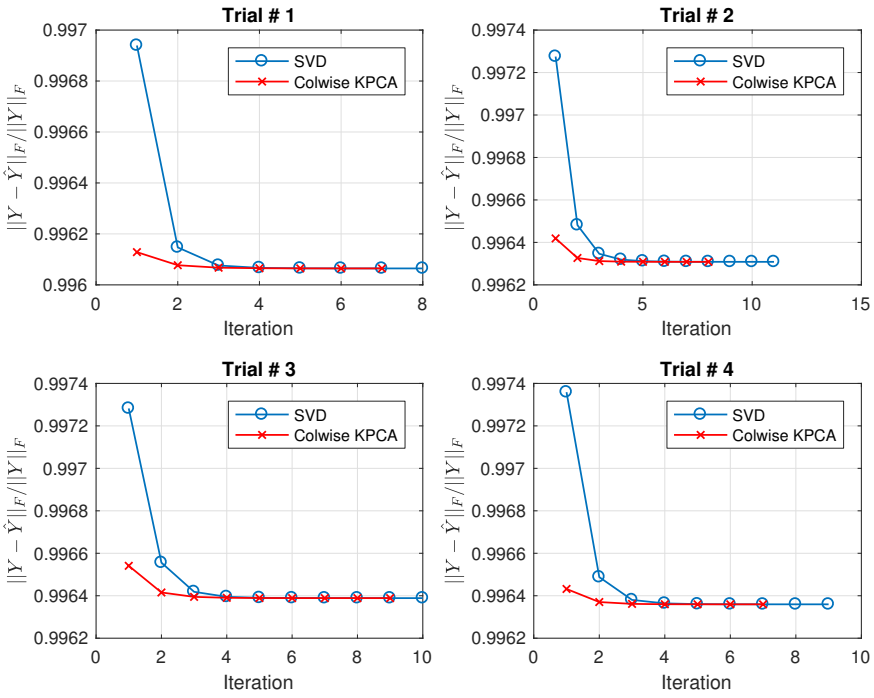


Figure 6.2: Plots of typical normalized RMSE, $\|\mathcal{Y} - \widehat{\mathcal{Y}}\|_F^2 / \|\mathcal{Y}\|_F^2$ versus iterations for four independent runs of the ALS algorithm, for a rank one signal. Here, $m_1 = 32$, $m_2 = 30$, $n = 80$, $\sigma_1 = 2.3$. ALS is terminated when the normalized RMSE dips below a threshold, or changes very little with iterations. When initialized with columnwise KPCA2, ALS converges in fewer iterations than the SVD to the same normalized RMSE.

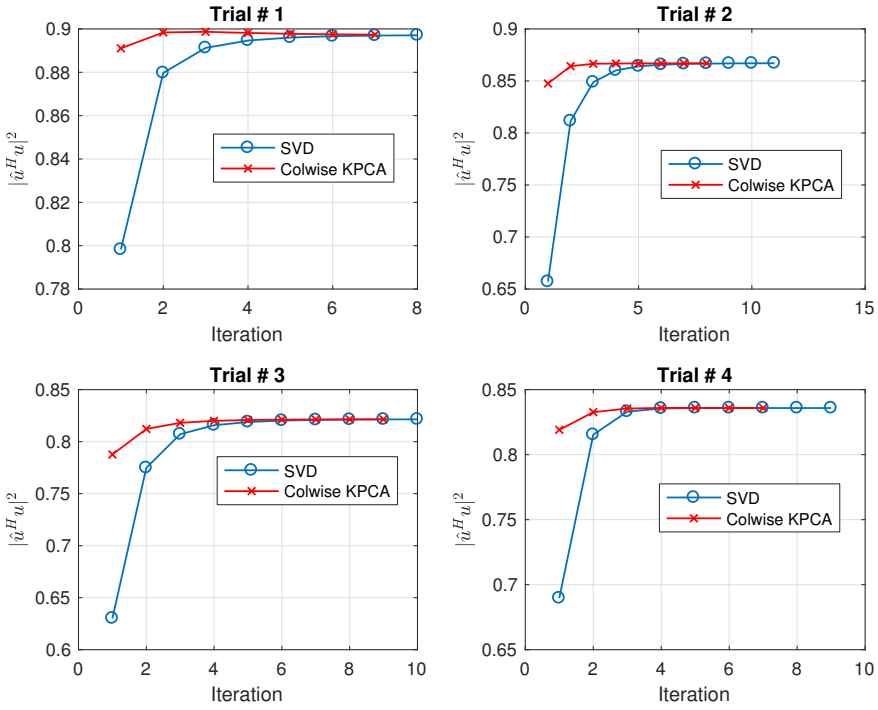


Figure 6.3: Plots of typical singular vector estimation accuracy, $|\widehat{u}_1^H u_1|^2$ versus iterations for the setup in Fig. 6.2. The estimation accuracy of $\widehat{u}_1$ initialized through Algorithm 6.3 is higher than when initialized using Algorithm 6.2 during the earlier iterations.

VI.4 New Weights for Low Rank Tensor Estimation

In this section, consider the problem (6.7), which is

$$z_{\mathsf{tsh}} = \arg\min_{z\in\mathbb{R}_+^r} \mathrm{SE}(z) := \arg\min_{z\in\mathbb{R}_+^r} \left\|\mathcal{S} - \widehat{\mathcal{S}}(z)\right\|_F^2.$$

We propose data driven algorithms that compute z_{tsh} approximately.

VI.4.1 TensorShrink for Rank One Signals

Consider the case of rank one signal.

Proposition VI.1. *Let $\mathcal{S} = \sigma_1 \bar{s}_1 \circ b_1 \circ a_1$, $\mathcal{Y} = \mathcal{S} + \mathcal{Z}$ as in (6.1), and*

$$\widehat{\mathcal{S}}(z) = z\,\widehat{\bar{s}}_1 \circ \widehat{b}_1 \circ \widehat{a}_1.$$

As in (6.8), let

$$u_1 := a_1 \otimes b_1, \quad \widehat{u}_1 := \widehat{a}_1 \otimes \widehat{b}_1.$$

and

$$z_{\mathsf{tsh}} = \arg\min_{z\in\mathbb{R}_+} \|\mathcal{S} - \widehat{\mathcal{S}}(z)\|_F^2.$$

Then,

$$z_{\mathsf{tsh}} = \sigma_1\, \Re\{(\widehat{a}_1^H a_1)\,(\widehat{b}_1^H b_1)\,(s_1^H \widehat{s}_1)\} = \frac{\sigma_1^2 |\widehat{u}_1^H u_1|^2 + \sigma_1 \Re\{(\widehat{u}_1^H u_1)(s_1^H Z \widehat{u}_1)\}}{\sigma_1^2 |\widehat{u}_1^H u_1|^2 + \widehat{u}_1^H Z Z^H \widehat{u}_1 + 2\sigma_1 \Re\{(\widehat{u}_1^H u_1)(s_1^H Z \widehat{u}_1)\}}. \tag{6.9}$$

Proof. We wish to solve the optimization problem

$$z_{\mathsf{tsh}} = \arg\min_{z\in\mathbb{R}_+} \|\mathcal{S} - z\,\widehat{\bar{s}}_1 \circ \widehat{b}_i \circ \widehat{a}_i\|_F^2.$$

By definition (6.3), Frobenius norm is invariant to a rearrangement of the elements. Unfolding the tensor $\mathcal{S}$ along mode 1 (see (3.3)) we have that

$$\begin{aligned} z_{\mathsf{tsh}} &= \arg\min_{z\in\mathbb{R}_+} \|\mathcal{S}_{[1]}^T - z\,(\widehat{a}_1 \otimes \widehat{b}_1)\,\widehat{s}_1^H\|_F^2 \\ &= \arg\min_{z\in\mathbb{R}_+} \|\sigma_1\,(a \otimes b)\,s_1^H - z\,(\widehat{a}_1 \otimes \widehat{b}_1)\,\widehat{s}_1^H\|_F^2 \end{aligned}$$

$$= \arg\min_{z \in \mathbb{R}_+} \sigma_1^2 + z^2 - 2z\, \Re\{(\widehat{a}_1 \otimes \widehat{b}_1)^H (a \otimes b)\, (s_1^H \widehat{s}_1)\}. \tag{6.10}$$

By differentiating (6.10) with respect to z, and solving the resulting equation, we obtain

$$z_{\mathsf{tsh}} = \sigma_1\, \Re\{(\widehat{a}_1^H a)\, (\widehat{b}_1^H b)\, (s_1^H \widehat{s}_1)\}. \tag{6.11}$$

Recall from Fig. 6.1 that $\widehat{s}_1 = Y^H \widehat{u}_1 / \|Y^H \widehat{u}_1\|_2$. Substituting for Y from (6.1),

$$\begin{aligned} s_1^H \widehat{s}_1 &= \frac{s_1^H Y^H \widehat{u}_1}{\|Y^H \widehat{u}_1\|_2} \\ &= \frac{s_1^H (Z^H + \sigma_1 s_1 u_1^H) \widehat{u}_1}{\widehat{u}_1^H (\sigma_1 u_1 s_1^H + Z)(\sigma_1 u_1 s_1^H + Z)^H \widehat{u}_1} \\ &= \frac{s_1^H Z^H \widehat{u}_1 + \sigma_1 (u_1^H \widehat{u}_1)}{\sigma_1^2 |\widehat{u}_1^H u_1|^2 + \widehat{u}_1^H Z Z^H \widehat{u}_1 + 2\sigma_1 \Re\{(\widehat{u}_1^H u_1)(s_1^H Z \widehat{u}_1)\}}. \end{aligned}$$

Plugging this into (6.11), we obtain expression (6.9) for z_{tsh}, which is the required result. □

Assumption VI.1. Assume if conditions of Proposition VI.1 hold, that as $m, n \to \infty$, $m/n \to c$,

$$s_1^H Z \widehat{u}_1 \xrightarrow{\text{a.s.}} 0.$$

Corollary VI.1. *Assume that the conditions of Proposition VI.1 hold, and Assumption VI.1 holds. Then we have that, as $m_1, m_2, n \to \infty$, $m_1 m_2 / n \to c$,*

$$z_{\mathsf{tsh}} \to \frac{\sigma_1^2 |\widehat{u}_1^H u_1|^2}{1 + \sigma_1^2 |\widehat{u}_1^H u_1|^2}. \tag{6.12}$$

Here too, as in Section V.3, we see that the optimal weight depends upon $|\widehat{u}_1^H u_1|^2$. However, $\widehat{u}_1$ is not computed via the SVD of Y, but via the ALS algorithm. Therefore, the asymptotic limit of $\widehat{z}_{\mathsf{tsh}}$ may be different from that in (5.25). The convergence in (6.12) also depends upon the nature of convergence of $|\widehat{u}_1^H u_1|^2$ to its limit. We present the data driven estimation of $\widehat{z}_{\mathsf{tsh}}$ in Section VI.5.

VI.4.2 TensorShrink for Higher Rank Signals

In this section, we consider the problem of deriving TensorShrink weights for a rank $r \geq 1$ signal.

Proposition VI.2. *Let Y be observed data as in (6.1), and let $\widehat{A} \in \mathbb{C}^{m_1 \times r}$, $\widehat{B} \in \mathbb{C}^{m_2 \times r}$, $\widehat{S} \in \mathbb{C}^{n \times r}$ estimates of A, B, S respectively, as in (6.8). Let z_{tsh} minimize the squared (denoising) error $\mathrm{SE}(z)$:*

$$z_{\mathsf{tsh}} = \arg\min_{z \in \mathbb{R}^r_+} \mathrm{SE}(z) := \arg\min_{z \in \mathbb{R}^r_+} \left\| \mathcal{S} - \sum_{i=1}^{r} z_i \, \widehat{s}_i \circ \widehat{b}_i \circ \widehat{a}_i \right\|_F^2 . \tag{6.13}$$

Then z_{tsh} is the solution to the system of equations

$$\begin{bmatrix} \Re\{\widehat{s}_1^H \widehat{s}_1 \widehat{u}_1^H \widehat{u}_1\} & \dots & \Re\{\widehat{s}_1^H \widehat{s}_r \widehat{u}_r^H \widehat{u}_1\} \\ & \ddots & \\ \Re\{\widehat{s}_r^H \widehat{s}_1 \widehat{u}_1^H \widehat{u}_r\} & \dots & \Re\{\widehat{s}_r^H \widehat{s}_r \widehat{u}_r^H \widehat{u}_r\} \end{bmatrix} \begin{bmatrix} z_1 \\ \vdots \\ z_r \end{bmatrix} = \begin{bmatrix} \Re\{\widehat{s}_1^H s_1 u_1^H \widehat{u}_1\} & \dots & \Re\{\widehat{s}_1^H s_r u_r^H \widehat{u}_1\} \\ & \ddots & \\ \Re\{\widehat{s}_r^H s_1 u_1^H \widehat{u}_r\} & \dots & \Re\{\widehat{s}_r^H s_r u_r^H \widehat{u}_r\} \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \vdots \\ \sigma_r \end{bmatrix} . \tag{6.14}$$

We will require the following lemma to prove this.

Lemma VI.1. *Given square matrices B, C, X of same dimensions, where X is a diagonal matrix,*

$$\frac{\partial}{\partial X} \mathrm{Tr}(XB) = B^T \odot I, \tag{6.15a}$$

$$\frac{\partial}{\partial X} \mathrm{Tr}(XBX^T C) = \left(C^T X B^T + CXB \right) \odot I, \tag{6.15b}$$

where $\odot$ denotes Hadamard product.

Proof. See [99]. □

Proof of Proposition VI.2. Consider the problem (6.13). Let

$$\Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_r), \quad Z = \mathrm{diag}(z_1, \dots, z_r). \tag{6.16}$$

Unfolding the tensors within the objective function along mode 1, we have that

$$\begin{aligned} \mathrm{SE}(z) : &= \left\| \mathcal{S}_{[1]} - \sum_{i=1}^{r} (\widehat{a}_i \otimes \widehat{b}_i) z_i \widehat{s}_i^H \right\|_F^2 \\ &= \left\| U \Sigma S^H - \widehat{U} Z \widehat{S}^H \right\|_F^2 \\ &= \mathrm{Tr}\left(\left(U \Sigma S^H - \widehat{U} Z \widehat{S}^H \right)^H \left(U \Sigma S^H - \widehat{U} Z \widehat{S}^H \right) \right) \end{aligned}$$

$$= \mathrm{Tr}\left(S\Sigma U^H U\Sigma S^H\right) + \mathrm{Tr}\left(\widehat{S}Z\widehat{U}^H\widehat{U}Z\widehat{S}^H\right) + 2\Re\left\{\mathrm{Tr}\left(\widehat{S}Z\widehat{U}^H U\Sigma S^H\right)\right\}. \tag{6.17}$$

The first term in the summation of (6.17) does not involve Z, and can be discarded for optimization. The quantity $\mathrm{SE}(z)$ is minimized at z which is the solution to

$$\underbrace{\frac{\partial}{\partial Z}\mathrm{Tr}\left(\widehat{S}Z\widehat{U}^H\widehat{U}Z\widehat{S}^H\right)}_{:=T_1} - \underbrace{\frac{\partial}{\partial Z}2\Re\left\{\mathrm{Tr}\left(\widehat{S}Z\widehat{U}^H U\Sigma S^H\right)\right\}}_{:=T_2} = 0. \tag{6.18}$$

Since the trace of a matrix is commutative, we have that

$$T_1 = \frac{\partial}{\partial Z}\mathrm{Tr}\left(Z\widehat{U}^H\widehat{U}Z\widehat{S}^H S\right), \tag{6.19a}$$

$$T_2 = \frac{\partial}{\partial Z}2\Re\left\{\mathrm{Tr}\left(Z\widehat{U}^H U\Sigma S^H\widehat{S}\right)\right\}. \tag{6.19b}$$

Now we use Lemma VI.1 to simplify T_1. In (6.15b), substitute

$$X = Z, \quad B = \widehat{U}^H\widehat{U}, \quad C = \widehat{S}^H S.$$

We obtain:

$$\begin{aligned} T_1 &= \frac{\partial}{\partial Z}\mathrm{Tr}\left(\widehat{S}Z\widehat{U}^H\widehat{U}Z\widehat{S}^H\right) \\ &= \left((\widehat{U}^H\widehat{U})^T Z(\widehat{S}^H\widehat{S})^T + (\widehat{U}^H\widehat{U})Z(\widehat{S}^H\widehat{S})\right)\odot I \\ &= 2\Re\{\left(\widehat{U}^H\widehat{U})Z(\widehat{S}^H\widehat{S})\right)\odot I\}. \end{aligned} \tag{6.20}$$

We now simplify the term $(\widehat{U}^H\widehat{U})Z(\widehat{S}^H\widehat{S})$ in (6.20). We have that

$$\begin{aligned} (\widehat{U}^H\widehat{U})Z(\widehat{S}^H\widehat{S}) &= \begin{bmatrix} \widehat{u}_1^H\widehat{u}_1 & \dots & \widehat{u}_1^H\widehat{u}_r \\ & \ddots & \\ \widehat{u}_r^H\widehat{u}_1 & \dots & \widehat{u}_r^H\widehat{u}_r \end{bmatrix} \begin{bmatrix} z_1 & & \\ & \ddots & \\ & & z_r \end{bmatrix} \begin{bmatrix} \widehat{s}_1^H\widehat{s}_1 & \dots & \widehat{s}_1^H\widehat{s}_r \\ & \ddots & \\ \widehat{s}_r^H\widehat{s}_1 & \dots & \widehat{s}_r^H\widehat{s}_r \end{bmatrix} \\ &= \sum_{j=1}^{r} z_j \begin{bmatrix} \widehat{u}_1\widehat{u}_j \\ \vdots \\ \widehat{u}_r^H\widehat{u}_j \end{bmatrix} \begin{bmatrix} \widehat{s}_j^H\widehat{s}_1 & \dots & \widehat{s}_j^H\widehat{s}_r \end{bmatrix} \end{aligned} \tag{6.21}$$

$$= \begin{bmatrix} \sum_{j=1}^{r} z_j \widehat{u}_1 \widehat{u}_j \widehat{s}_j \widehat{s}_1 & * & * \\ * & & * \\ * & * & \sum_{j=1}^{r} z_j \widehat{u}_r \widehat{u}_j \widehat{s}_j \widehat{s}_r \end{bmatrix} \tag{6.22}$$

Substituting (6.22) in (6.20), we see that T_1 is given by

$$T_1 = 2 \begin{bmatrix} \sum_{j=1}^{r} z_j \Re\{\widehat{u}_1 \widehat{u}_j \widehat{s}_j \widehat{s}_1\} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \sum_{j=1}^{r} z_j \Re\{\widehat{u}_r \widehat{u}_j \widehat{s}_j \widehat{s}_r\} \end{bmatrix}. \tag{6.23}$$

Now consider T_2 in (6.19b). Use Proposition VI.1: in (6.15b), substitute

$$X = Z, \quad B = \widehat{U}^H U \Sigma S^H \widehat{S}.$$

From computations similar to (6.21) through (6.22), we see that

$$T_2 = 2 \begin{bmatrix} \sum_{j=1}^{r} \sigma_j \Re\{\widehat{s}_1 s_j u_j \widehat{u}_1\} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \sum_{j=1}^{r} \sigma_j \Re\{\widehat{s}_r s_j u_j \widehat{u}_r\} \end{bmatrix}. \tag{6.24}$$

Since z is the solution to the system of equations $T_1 = T_2$, it is the solution to

$$\begin{bmatrix} \sum_{j=1}^{r} z_j \Re\{\widehat{u}_1 \widehat{u}_j \widehat{s}_j \widehat{s}_1\} \\ \vdots \\ \sum_{j=1}^{r} z_j \Re\{\widehat{u}_r \widehat{u}_j \widehat{s}_j \widehat{s}_r\} \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{r} \sigma_j \Re\{\widehat{s}_1 s_j u_j \widehat{u}_1\} \\ \vdots \\ \sum_{j=1}^{r} \sigma_j \Re\{\widehat{s}_r s_j u_j \widehat{u}_r\} \end{bmatrix}. \tag{6.25}$$

Expressing (6.25) in matrix-vector form gives us (6.14), the required result. □

Note that all terms in the matrix on the left hand side of (6.14) can be computed directly from the data. The quantities which must be characterized to determine the optimal weights, $\Re\{\widehat{s}_i^H s_j u_j^H \widehat{u}_i\}$, involve the inner product between the estimated, and corresponding underlying singular vectors.

In general, the characterization of these quantities remains an open problem. In the special case when U, $\widehat{U}$ contain orthonormal columns, and $\widehat{U}$ contains top r left singular vectors of Y, the solution reduces to

$$z_{\mathsf{tsh,i}} = \sigma_i \, \Re\{\widehat{s}_i^H s_i u_i^H \widehat{u}_i\} \xrightarrow{\text{a.s.}} -2\frac{D_{\mu_Z}(\rho_i)}{D'_{\mu_Z}(\rho_i)}, \tag{6.26}$$

where $D_{\mu_Z}(\cdot)$, $D'_{\mu_Z}(\cdot)$ are defined in (5.24c), (5.24d). These are exactly the OptShrink weights derived by Nadakuditi [15]. In the case where $\widehat{U}$ contains estimates derived using an algorithm other than SVD, we are required to analyze $\Re\{\widehat{s}_i^H s_j u_j^H \widehat{u}_i\}$ to determine TensorShrink weights. For example, in the case of estimation algorithm described by Fig. 6.1, we analyzed this term for the rank one signal case, resulting in the expression (6.12).

The key idea uncovered here is that the optimal weights for signal denoising, z_{tsh} are a shrinkage of the coefficients $\sigma_1, \ldots, \sigma_r$. Since finding the exact shrinkage weights is currently an open problem, we propose using suboptimal shrinkage weights which are higher rank analogues of optimum weight for rank one signal given by (6.12). Specifically, we propose using the following suboptimal weights, for $i = 1, \ldots, r$:

$$z_{\mathsf{tsh,i}} = \frac{\sigma_i^2 |\widehat{u}_i^H u_i|^2}{1 + \sigma_i^2 |\widehat{u}_i^H u_i|^2}. \tag{6.27}$$

VI.5 Data Driven Estimation of $|\widehat{u}_i^H u_i|^2$ and TensorShrink Algorithm

From the expressions for TensorShrink weights (6.12) and (6.27), we see that the TensorShrink weights depend upon $|\widehat{u}_i^H u_i|^2$. In the case of estimation using SVD, and orthonormal $\{u_i\}_{i=1}^r$, this quantity is characterized exactly in [14]. In this section, we propose an algorithm based on an approximation of $|\widehat{u}_i^H u_i|^2$, for $\{\widehat{u}_i\}_{i=1}^r$ output from the ALS algorithm.

The procedure is detailed in Algorithm 6.4. We outline the steps in detail below. Recall

from Fig. 6.1 that $\widehat{u}_i = \widehat{a}_i \otimes \widehat{b}_i$ for $i = 1, \ldots, r$. Therefore,

$$|\widehat{u}_i^H u_i|^2 = |(\widehat{a}_i \otimes \widehat{b}_i)^H (\widehat{a}_i \otimes \widehat{b}_i)|^2 = |\widehat{a}_i^H a_i|^2 \, |\widehat{b}_i^H b_i|^2.$$

In other words, we need to approximate the accuracy of $\widehat{a}_i$, $\widehat{b}_i$ with respect to their underlying vectors from data.

The characterization of inner-product-squared terms $|\widehat{a}_i^H a_i|^2$, $|\widehat{b}_i^H b_i|^2$ for $\{\widehat{a}_i\}_{i=1}^r$, $\{\widehat{b}_i\}_{i=1}^r$ output from the ALS algorithm remains an open problem. Instead, we exploit our choice of column-wise KPCA based initialization for the ALS algorithm, Algorithm 6.3, and use approximations of the quantities $|\widehat{a}_{\mathsf{ckpca,i}}^H a_i|^2$, $|\widehat{b}_{\mathsf{ckpca,i}}^H b_i|^2$. Recall that $\widehat{a}_{\mathsf{ckpca,i}}$, $\widehat{b}_{\mathsf{ckpca,i}}$ are respectively the principal left and right singular vectors of $\mathcal{R}_{m\times 1\to m_1\times m_2}(\widehat{u}_{\mathsf{svd,i}})$, the matrix obtained by a Van Loan-Pitsianis rearrangement of the i^{th} principal left singular vector of the data.

This approach is motivated by Fig. 6.3, which tells us inner-product-squared $|\widehat{u}_i^H u_i|^2$ after the ALS algorithm has converged is at least as large as it was when it was initialized. Therefore, the approximations of $|\widehat{a}_{\mathsf{ckpca,i}}^H a_i|^2$, $|\widehat{b}_{\mathsf{ckpca,i}}^H b_i|^2$ serve as a lower bound for $|\widehat{u}_i^H u_i|^2$. A further justification of this approach is provided by Fig. 6.4.

Steps 1–6 of Algorithm 6.4 are a direct consequence of Steps 1–3 in Algorithm 6.3. In Step 7, we approximate σ_i^2 in (6.27) via the D-transform. For Steps 8–12, consider the SVD of $\widehat{U}_{\mathcal{R},i} = \mathcal{R}_{m\times 1\to m_1\times m_2}(\widehat{u}_{\mathsf{svd,i}})$:

$$\widehat{U}_{\mathcal{R},i} = \begin{bmatrix} \widehat{a}_{\mathsf{ckpca,i}} & \widehat{A}_i^{\perp} \end{bmatrix} \begin{bmatrix} \sigma_{\mathcal{R},s} & \\ & \Sigma_{\mathcal{R},n} \end{bmatrix} \begin{bmatrix} \widehat{b}_{\mathsf{ckpca,i}} & \widehat{B}_i^{\perp} \end{bmatrix}^H.$$

Motivated by [14, Theorem 2.10], we approximate

$$|\widehat{a}_i^H a_i|^2 \approx |\widehat{a}_{\mathsf{ckpca,i}}^H a_i|^2 \approx -2\frac{\varphi(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n}) D(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}{D'(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}, \tag{6.28a}$$

$$|\widehat{b}_i^H b_i|^2 \approx |\widehat{b}_{\mathsf{ckpca,i}}^H b_i|^2 \approx -2\frac{\widetilde{\varphi}(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n}) D(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}{D'(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}, \tag{6.28b}$$

where φ, $\widetilde{\varphi}$, D, and D' functions are defined in equations (5.24). Hence,

$$\begin{aligned} |\widehat{u}_i^H u_i|^2 &= |\widehat{a}_i^H a_i|^2 |\widehat{b}_i^H b_i|^2 \approx |\widehat{a}_{\mathsf{ckpca,i}}^H a_i|^2 |\widehat{b}_{\mathsf{ckpca,i}}^H b_i|^2 \\ &\approx -2\frac{\varphi(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n}) D(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}{D'(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})} \cdot -2\frac{\widetilde{\varphi}(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n}) D(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}{D'(\sigma_{\mathcal{R},\mathsf{s}}, \Sigma_{\mathcal{R},n})}. \end{aligned} \tag{6.29}$$

Finally, in Step 13, we compute $\widehat{z}_{\mathsf{tsh,i}}$ using (6.27).

Algorithm 6.4: TensorShrink for Order Three Tensors

Input: $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, data tensor

Input: r, guess for signal rank

Output: $\widehat{z}_{\mathsf{tsh}} \in \mathbb{R}^r_+$, TensorShrink weights

1 Compute $Y = \mathcal{Y}_{[1]}$, unfolded tensor

2 Compute $Y = \widehat{U}_{\mathsf{svd}} \widehat{\Sigma}_{\mathsf{svd}} \widehat{V}^H_{\mathsf{svd}}$, the SVD of Y

3 Let $\widehat{\Sigma}_{\mathsf{svd,s}} = \widehat{\Sigma}_{\mathsf{svd}}(1:r, 1:r)$, $\widehat{\Sigma}_{\mathsf{svd,n}} = \widehat{\Sigma}_{\mathsf{svd}}(r+1:m, r+1:n)$

4 **for** $i = 1$ **to** r **do**

5 Let $\widehat{\sigma}_{\mathsf{svd,i}} = \widehat{\Sigma}_{\mathsf{svd,s}}(i,i)$

6 Let $\widehat{u}_{\mathsf{svd,i}} = \widehat{U}_{\mathsf{svd}}(:,i)$

7 Compute $\widehat{\sigma}_i^2 = \dfrac{1}{D\left(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_{\mathsf{svd,n}}\right)}$ where D is defined in (5.24c)

8 Compute $\widehat{U}_{\mathcal{R},i} = \mathcal{R}_{m_1\, m_2 \times 1 \to m_1 \times m_2}(\widehat{u}_{\mathsf{svd,i}})$

9 Compute $\widehat{\Sigma}_{\mathcal{R},i}$ = (diagonal matrix of) singular values of $\widehat{U}_{\mathcal{R},i}$

10 Let $\widehat{\sigma}_{\mathcal{R},\mathsf{s}} = \widehat{\Sigma}_{\mathcal{R}}(1,1)$, $\widehat{\Sigma}_{\mathcal{R},n} = \widehat{\Sigma}_{\mathcal{R}}(2:m_1, 2:m_2)$

11 Approximate $|\widehat{a}_i^H a_i|^2$, $|\widehat{b}_i^H b_i|^2$ using (6.28a), (6.28b)

12 Approximate $|\widehat{u}_i^H u_i|^2$ using (6.29)

13 Compute $\widehat{z}_{\mathsf{tsh,i}}$ using (6.27)

14 **end**

15 Sort $\{\widehat{z}_{\mathsf{tsh,i}}\}$ in descending order

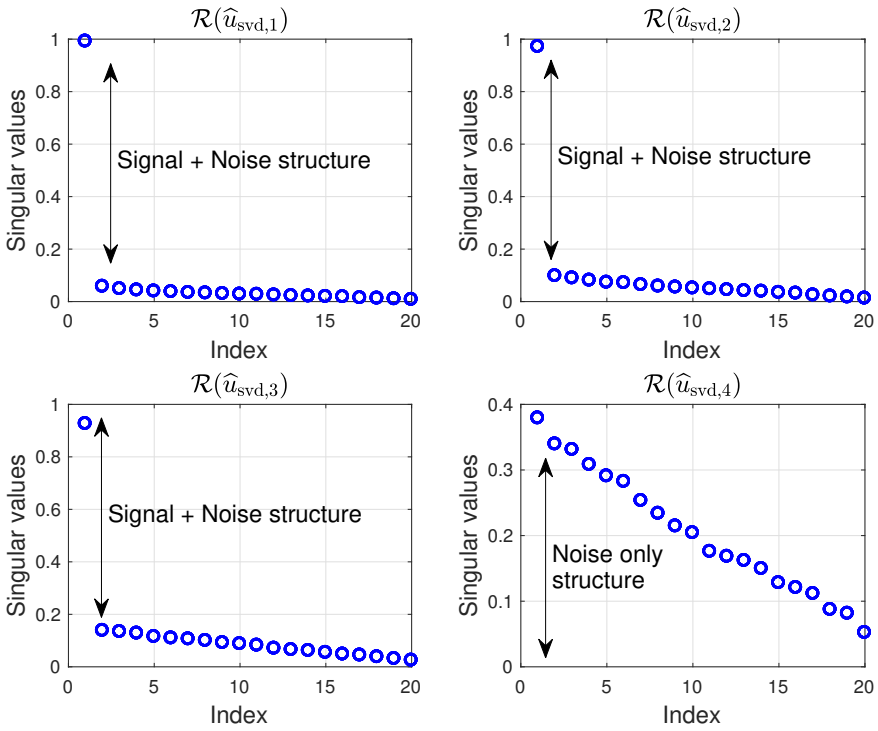


Figure 6.4: Spectrum of $\widehat{U}_{\mathcal{R},i} = \mathcal{R}(\widehat{u}_{\text{svd,i}})$, *i.e.*, the matrices resulting from the Van Loan-Pitsianis rearrangement of the principal four singular vectors, for a system with a rank 3 signal. Recall that these singular vectors are used to initialize the ALS iterations: the principal left and right singular vectors of $\mathcal{R}(\widehat{u}_{\text{svd,i}})$ give $\widehat{a}_i$ and $\widehat{b}_i$ respectively for the initial iteration. We see that the spectra of the top 3 singular vectors have a signal-plus-noise structure, whereas the spectrum of $\mathcal{R}(\widehat{u}_{\text{svd,4}})$ is noise-like. This justifies approximating $|\widehat{a}_i^H a_i|^2$ and $|\widehat{b}_i^H b_i|^2$ using the spectra of $\mathcal{R}(\widehat{u}_{\text{svd,i}})$.

VI.6 TensorBeamshrink: Beamforming via TensorShrink Based Weights

In this section, we consider the problem of beamforming for clutter suppression in a STAP system. We propose an algorithm called TensorBeamshrink towards computing a beamformer via a low rank tensor estimation based approach.

Recall from (5.2) in Section V.2 that the observed data in this system can be modeled as

$$Y = \sum_{i=1}^{r} \sigma_i \, u_i \, s_i^H + Z, \tag{6.30}$$

where $\{\sigma_i\}_{i=1}^r \in \mathbb{R}_+$ are the clutter powers, $\{u_i\}_{i=1}^r \in \mathbb{C}^m$ are the (unit norm) steering vectors associated with clutter locations, $\{s_i\}_{i=1}^r \in \mathbb{C}^n$ are the (unit norm) signal vectors, and Z is the noise matrix, independent of the signal, whose elements have 0 mean and covariance $1/n$. Furthermore, recall from (3.51) that the clutter steering vector has a double Kronecker product structure:

$$u_i = a_i \otimes b_i, \tag{6.31}$$

where $a_i \in \mathbb{C}^{m_1}$ is a unit norm vector parameterized by the Doppler shift at clutter location, and $b_i \in \mathbb{C}^{m_2}$ is a unit norm vector parameterized by the clutter's angular location.

We may view matrix Y in (6.30) as a rearranged tensor. As we showed in (3.3) of Section III.2.1, if we unfold the observed tensor $\mathcal{Y}$ of (6.1) along mode 1, we get

$$\begin{aligned} \mathcal{Y}_{[1]}^T &= \sum_{i=1}^{r} \sigma_i \underbrace{(a_i \otimes b_i)}_{=u_i} s_i^H + \mathcal{Z}_{[1]} \\ &= Y. \end{aligned}$$

Thus, the STAP data model of (6.30), and low rank tensor signal plus noise model of (6.1) are equivalent up to a rearrangement operation.

Recall from Section V.2 that the goal of STAP is to compute a beamformer from observed data $Y = \mathcal{Y}_{[1]}$. Specifically, we consider the DMR beamformer, which is given by (5.4):

$$w_{\text{orc}} = \frac{R^{-1} \, u_t}{\|R^{-1} \, u_t\|_F}$$

where $R = \mathbb{E}\left[YY^H\right]$ is the population covariance matrix of observed data. For Y as in (6.30),

$$R = I + \sum_{i=1}^{r} \sigma_i \underbrace{(a_i \otimes b_i)}_{=u_i} \underbrace{(a_i \otimes b_i)^H}_{=u_i^H}. \tag{6.32}$$

Using a low rank tensor estimation based approach, R is canonically estimated via the ALS algorithm:

$$\widehat{R}_{\mathsf{als}} = I + \sum_{i=1}^{r} \widehat{\sigma}_{\mathsf{als,i}}\, \widehat{u}_{\mathsf{als,i}}\, \widehat{u}_{\mathsf{als,i}}^H, \tag{6.33}$$

where $\widehat{\sigma}_{\mathsf{als,i}}$ and $\widehat{u}_{\mathsf{als,i}}$ are estimated using the ALS algorithm as shown in Fig. 6.1. We can thus form an ALS based beamformer estimate:

$$\widehat{w}_{\mathsf{als}} = \frac{\widehat{R}_{\mathsf{als}}^{-1}\, u_t}{\|\widehat{R}_{\mathsf{als}}^{-1}\, u_t\|_F}. \tag{6.34}$$

In (6.33), $\widehat{\sigma}_{\mathsf{als,i}}$ may be interpreted as a weight associated with the matrix $\widehat{u}_{\mathsf{als,i}}\, \widehat{u}_{\mathsf{als,i}}^H$. In Section V.3, we derived the optimum weights for SVD based beamformers. From (5.13), (5.23) and (5.25), these weights (which we called BeamShrink weights) had the form

$$z_{\mathsf{bsh,i}} = \frac{\sigma_i^2 |\widehat{u}_{\mathsf{svd,i}}^H u|^2}{1 + \sigma_i^2 (1 - |\widehat{u}_{\mathsf{svd,i}}^H u|^2)} \quad \text{for } i = 1, \ldots, r. \tag{6.35}$$

In this section, we propose a similar re-weighting for the ALS based beamformer given by (6.34). The derivation of exact optimal weights as in Section V.3 remains an open problem; we cannot reuse the derivation in Section V.3 since the set of vectors $\{u_i\}_{i=1}^r$ and $\{\widehat{u}_i\}_{i=1}^r$ are not orthonormal. However, inspired by (6.35) we propose the following suboptimal weighting, which we call TensorBeamshrink weights:

$$z_{\mathsf{tbsh,i}} = \frac{\sigma_i^2 |\widehat{u}_i^H u|^2}{1 + \sigma_i^2 (1 - |\widehat{u}_i^H u|^2)} \quad \text{for } i = 1, \ldots, r, \tag{6.36}$$

where $\{\widehat{u}_i\}_{i=1}^r$ are ALS based estimates of $\{u_i\}_{i=1}^r$, computed as outlined by Fig. 6.1.

By comparing equations (6.36) and (6.27), we see that the TensorShrink and TensorBeamshrink weights involve the same terms: σ_i, and $|\widehat{u}_i^H u|^2$ for $i = 1, \ldots, r$. Therefore, the TensorBeamshrink weights of (6.36) may be estimated from data in a manner analogous to to the computation of TensorShrink weights, *i.e.*, via equations (6.28a), (6.28b), and (6.29).

We do so in Algorithm 6.5, which we call the TensorBeamshrink algorithm.

Algorithm 6.5: TensorBeamshrink for Order Three Tensors

Input: $\mathcal{Y} \in \mathbb{C}^{n \times m_1 \times m_2}$, data tensor

Input: r, guess for signal rank

Output: $\widehat{z}_{\mathsf{tbsh}} \in \mathbb{R}_+^r$, TensorShrink weights

Compute $Y = \mathcal{Y}_{[1]}$, unfolded tensor

Compute $Y = \widehat{U}_{\mathsf{svd}} \widehat{\Sigma}_{\mathsf{svd}} \widehat{V}_{\mathsf{svd}}^H$, the SVD of Y

Let $\widehat{\Sigma}_{\mathsf{svd,s}} = \widehat{\Sigma}_{\mathsf{svd}}(1:r, 1:r)$, $\widehat{\Sigma}_{\mathsf{svd,n}} = \widehat{\Sigma}_{\mathsf{svd}}(r+1:m, r+1:n)$

for $i = 1$ **to** r **do**

  Let $\widehat{\sigma}_{\mathsf{svd,i}} = \widehat{\Sigma}_{\mathsf{svd,s}}(i,i)$

  Let $\widehat{u}_{\mathsf{svd,i}} = \widehat{U}_{\mathsf{svd}}(:,i)$

  Compute $\widehat{\sigma}_i^2 = \dfrac{1}{D\left(\widehat{\sigma}_{\mathsf{svd,i}}, \widehat{\Sigma}_{\mathsf{svd,n}}\right)}$ where D is defined in (5.24c)

  Compute $\widehat{U}_{\mathcal{R},i} = \mathcal{R}_{m_1\, m_2 \times 1 \to m_1 \times m_2}(\widehat{u}_{\mathsf{svd,i}})$

  Compute $\widehat{\Sigma}_{\mathcal{R},i}$ = (diagonal matrix of) singular values of $\widehat{U}_{\mathcal{R},i}$

  Let $\widehat{\sigma}_{\mathcal{R},\mathsf{s}} = \widehat{\Sigma}_{\mathcal{R}}(1,1)$, $\widehat{\Sigma}_{\mathcal{R},n} = \widehat{\Sigma}_{\mathcal{R}}(2:m_1, 2:m_2)$

  Approximate $|\widehat{a}_i^H a_i|^2$, $|\widehat{b}_i^H b_i|^2$ using (6.28a), (6.28b)

  Approximate $|\widehat{u}_i^H u_i|^2$ using (6.29)

  Compute $\widehat{z}_{\mathsf{tbsh,i}}$ using (6.36)

end

Sort $\{\widehat{z}_{\mathsf{tbsh,i}}\}$ in descending order

Given TensorBeamshrink weights, we can form the TensorBeamshrink beamformer as

$$\widehat{w}_{\mathsf{tbsh}} = \frac{\widehat{R}_{\mathsf{tbsh}}^{-1} u_t}{\|\widehat{R}_{\mathsf{tbsh}}^{-1} u_t\|_2}, \tag{6.37}$$

where

$$\widehat{R}_{\mathsf{tbsh}} = I + \sum_{i=1}^{r} z_{\mathsf{tbsh,i}}^2 \, \widehat{u}_i \, \widehat{u}_i^H,$$

in which $\{\widehat{u}_i\}_{i=1}^r$ are computed as in Fig. 6.1.

VI.7 Numerical Simulations

Figure 6.5 plots the average denoising error, $\|\mathcal{S} - \widehat{\mathcal{S}}(z)\|_F$, for a tensor of order three with $m_1 = 20$, $m_2 = 30$, $n = 400$, over 500 trials. The signal has rank 30. However, it has an effective rank of 19, where

$$\text{effective rank} \;=\; \text{number of } \sigma_i > (m/n)^{1/4}.$$

Effective rank quantifies the number of informative principal components in the SVD of Y. We expect that the denoising error increases when input rank is greater than effective rank, as our estimated signal tensor will then contain uninformative, or noisy, components. "ALS" refers to when the weights are determined by the ALS algorithm, *i.e.*, $z_i = \widehat{\sigma}_i$, as described by Algorithm 6.1, "TensorShrink" refers to when weights are computed using Algorithm 6.4, and "Exact solution" refers to when weights are obtained by solving the optimization problem

$$z_{\mathsf{fmn}} = \|\mathcal{S} - \widehat{\mathcal{S}}(z)\|_F \tag{6.38}$$

numerically in `MATLAB`. Note that the exact solution weights require knowledge of the true signal, $\mathcal{S}$. This plot shows that we achieve better denoising using TensorShrink weights relative to the weights returned by the ALS algorithm.

Fig. 6.7 plots the ALS weight, TensorShrink weight, and exact solution weight vs σ_i for a rank two system, averaged over 100 trials. The plots highlight the non-convex nature of the shrinkage operation: at low SNRs, TensorShrink weights are lower, indicating greater shrinkage of the estimated component.

Fig. 6.8a plots the beampattern due the ALS algorithm based beamformer of (6.34), and Fig. 6.8b plots the beampattern due to the TensorBeamshrink based beamformer of (6.37), for a system with $m = 600$, $n = 400$, and a rank one signal. The input rank to the ALS algorithm is 20, so that the rank has been overestimated. We see that using TensorBeamshrink mitigates the effect of rank overestimation.

Fig. 6.9a plots the average difference in SINR over 100 trials, between the Tensor-Beamshrink based beamformer of (6.37) and the ALS based beamformer of (6.34) for the system in Fig. 6.8a. We see that using TensorBeamshrink gives us SINR gain of $+0.08$ dB

at target location relative to the ALS. Fig. 6.9b plots the average difference in SINR over 100 trials, between theTensorBeamshrink based beamformer of (6.37) and the SVD based beamformer of (5.10) for the system in Fig. 6.8a. We see that using TensorBeamshrink gives us SINR gain of +0.28 dB at target location relative to the SVD.

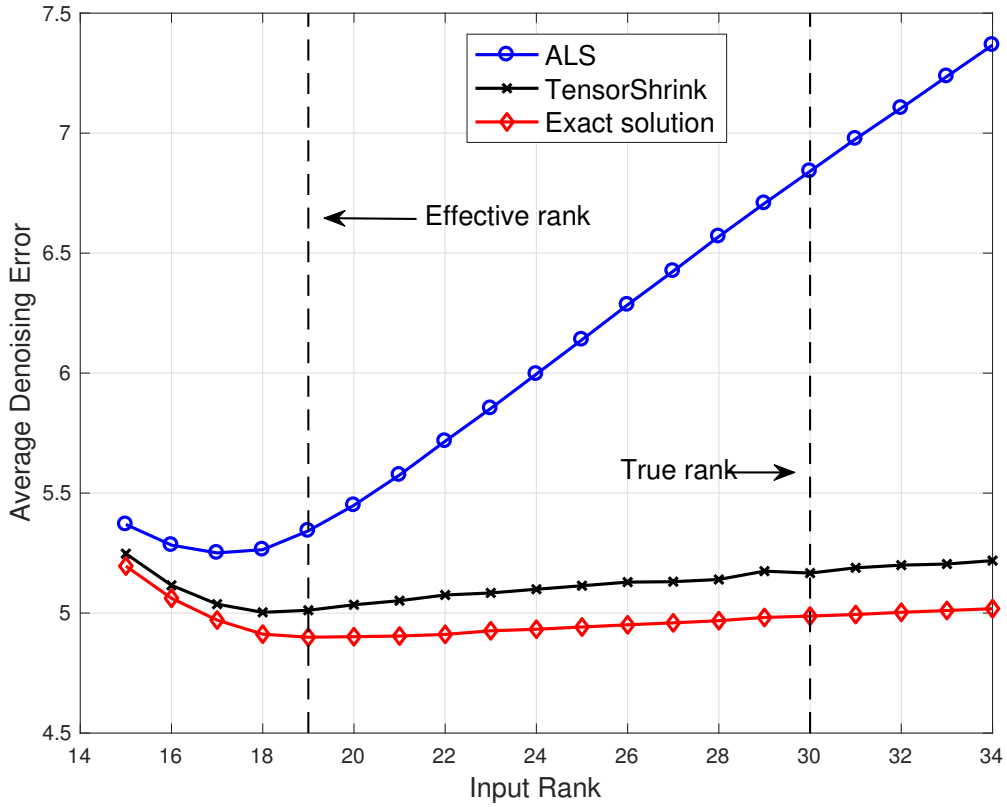


Figure 6.5: Plot of denoising error vs. input rank, for a system with $m_1 = 20$, $m_2 = 30$ and $n = 400$, over 500 trials. The error of ALS algorithm increases sharply when the input rank is greater than the effective rank, due to the addition of uninformative components to the signal estimate. The use of TensorShrink weights mitigates performance loss due to rank overestimation, and yields much lower denoising error, comparable to weights computed numerically using the oracle signal.

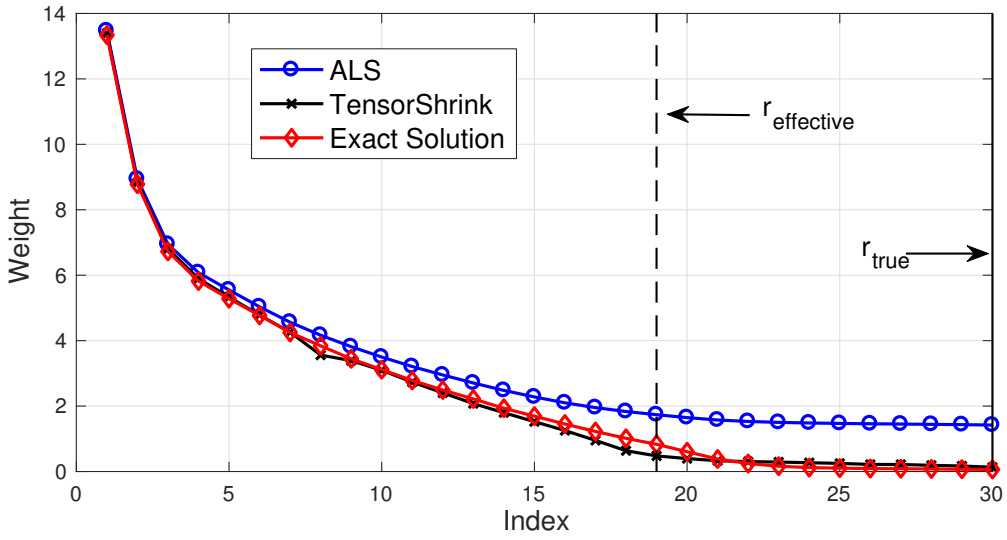
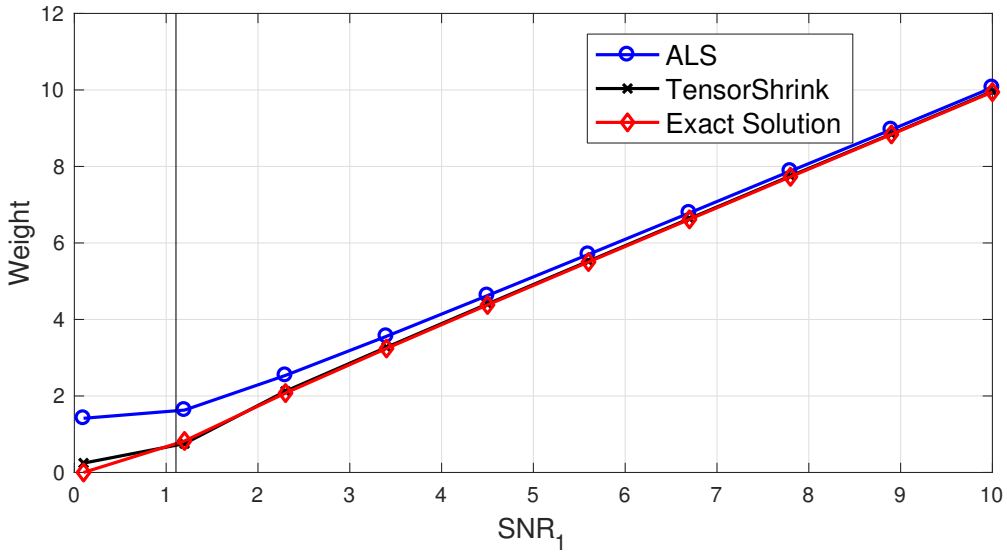
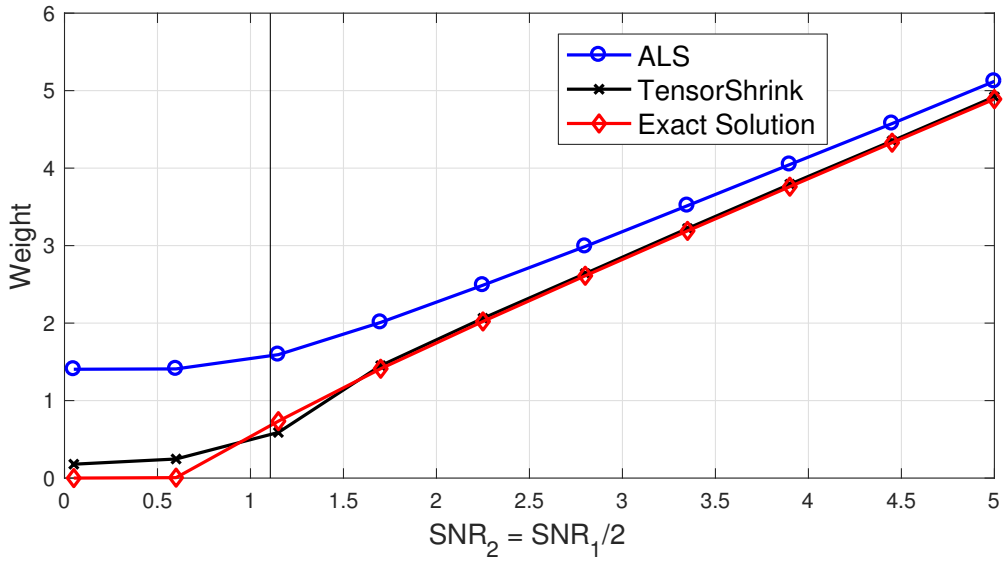


Figure 6.6: Average weights over 500 trials returned by the ALS and TensorShrink algorithms, compared to the optimum weights computed numerically. The input rank is 30, which is the true rank. We see that the smaller weights are shrunk much more: the lowest weights are close to 0. This gap between the ALS weights and TensorShrink weights leads to the gain in MSE performance at rank = 30 in Fig. 6.5.

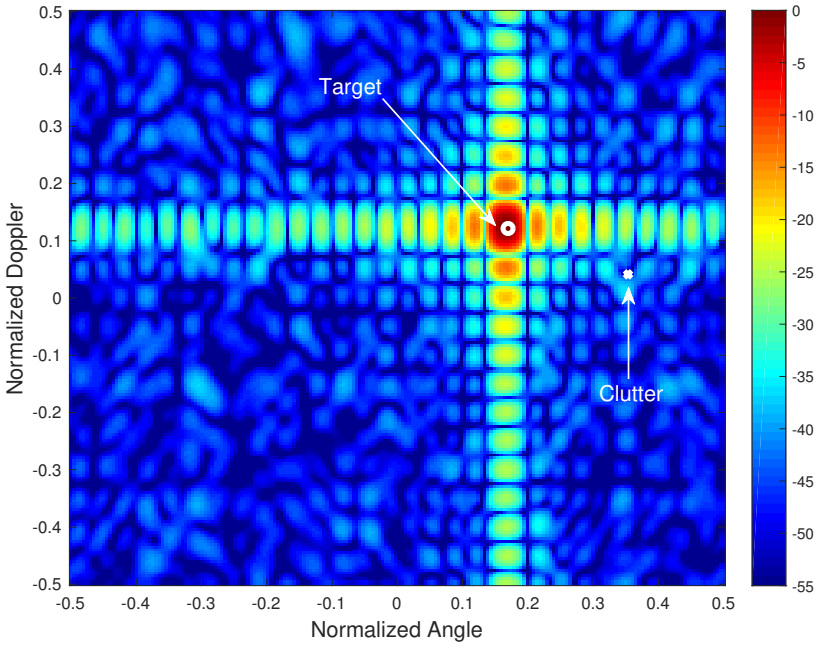


(a) Average weight $\widehat{\sigma}_1$ returned by ALS, $\widehat{z}_{\mathsf{tsh},1}$ returned by TensorShrink, and $z_{\mathsf{fmn},1}$ found by solving (6.38) numerically vs σ_1 for a system with $m_1 = 20$, $m_2 = 30$, $n = 400$. The signal has rank two, and $\sigma_2 = \sigma_1/2$. This plot illustrates the nature of the shrinkage. At low σ_1, the estimates returned by ALS are uninformative, so $\widehat{z}_{\mathsf{tsh}}$ is lower, causing higher shrinkage of the estimated component. At high SNRs, the TensorShrink weight is close to the ALS weight, indicating that no shrinkage is necessary for denoising.

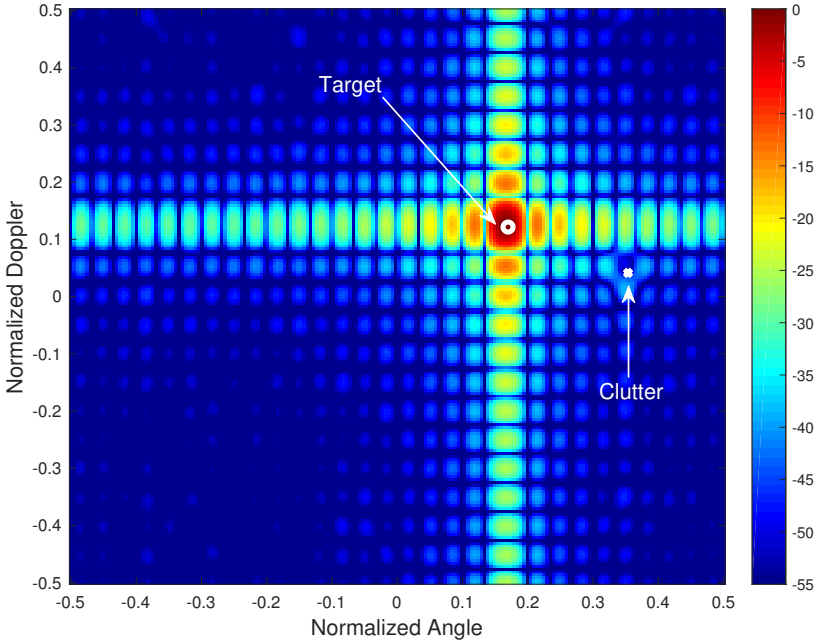


(b) Average weights $\widehat{\sigma}_2$, $\widehat{z}_{\mathsf{tsh},2}$ and $\widehat{z}_{\mathsf{tsh},2}$, corresponding to the weaker source. The effect of shrinkage at lower SNR is pronounced. Particularly, note that below $\sigma_{\mathsf{crit}} = (m/n)^{1/4}$, the point below which SVD estimates are completely uninformative, the weight is very close to 0.

Figure 6.7: TensorShrink weights vs. σ_i for a rank two system.

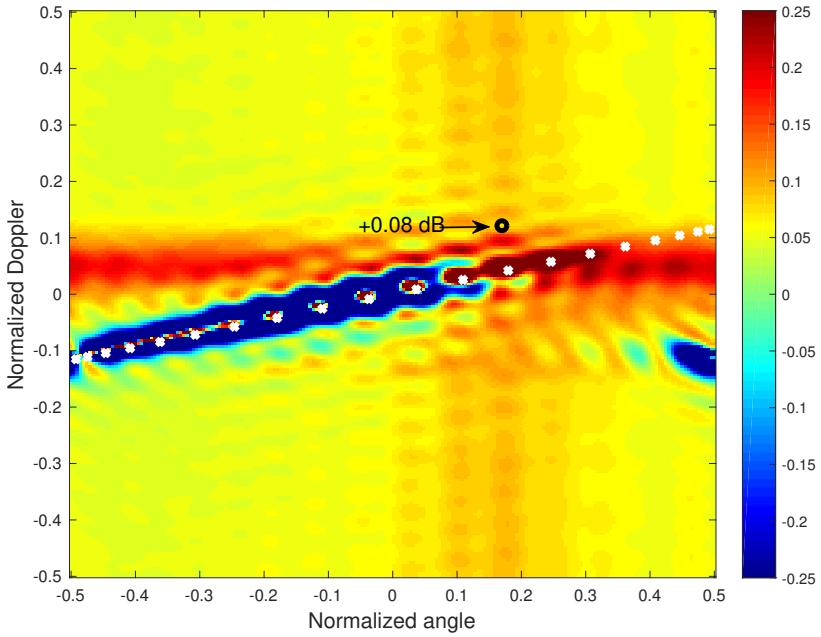


(a) Beampattern using the ALS based beamformer of (6.34) for a system with a single source, with $m = 600$, $n = 400$, with input rank 20. We see that the performance of the beamformer has degraded severely due to rank overestimation.

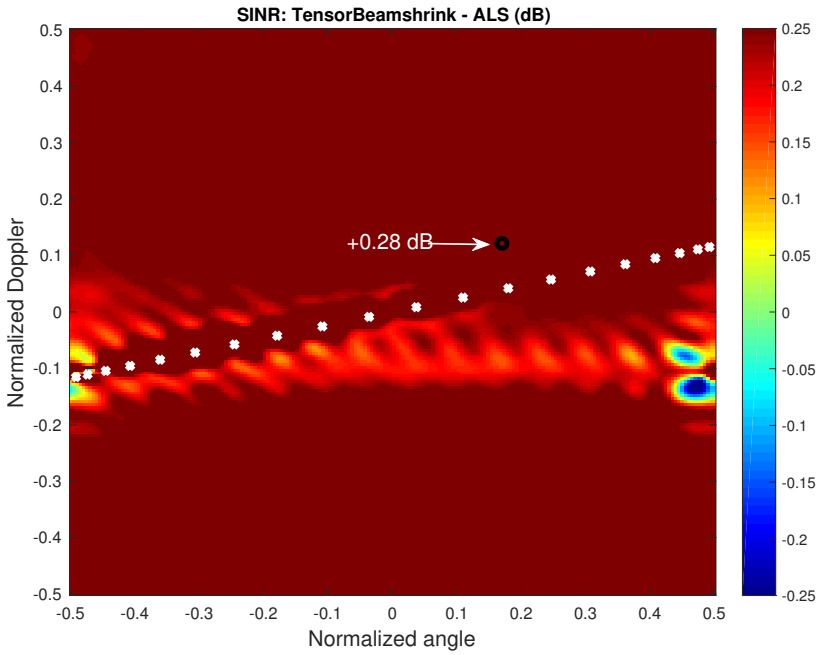


(b) Beampattern of the TensorBeamshink based beamformer of (6.37) for the same system in Fig. 6.8a. We see that using TensorBeamshrink mitigates the effect of rank overestimation.

Figure 6.8: Beampatterns of beamformers computed using weights returned by the ALS algorithm and TensorBeamshrink.



(a) Average difference in SINR due to the TensorBeamshrink based beamformer of (6.37) and the regular ALS based beamformer of (6.34), over 100 trials, for the system of Fig. 6.8a. We see that TensorBeamshrink gives us +0.08 dB gain at target relative to regular ALS.



(b) Average difference in SINR due to the TensorBeamshrink based beamformer of (6.37) and the regular SVD based beamformer of (5.10), over 100 trials, for the system of Fig. 6.8a. We see that using TensorBeamshrink gives +0.28 dB gain at target relative to the SVD based beamformer.

Figure 6.9: SINR gain due to TensorBeamshrink relative to the ALS and SVD based beamformers.

VI.8 Conclusions

In this chapter, we studied the estimation of low rank tensors. We derived expressions for optimal weights to reduced denoising error in low rank tensor estimates. The characterization of these terms remains an open problem. However, we were able to exploit the nature of these expressions to design an algorithm for suboptimal denoising, which we called TensorShrink. We showed that re-weighting the output of the ALS algorithm using TensorShrink leads to lower denoising error using simulations. We combined the idea of Tensorshrink with Beamshrink of Chapter V to derive TensorBeamshrink weights, which led to improved performance of ALS based beamformers in STAP. The characterization of terms involved in determining exact optimal weights remains an important open problem.

CHAPTER VII

Afterword and Future Work

In Chapter II, we characterized the performance of the MUSIC algorithm for a single source, in the presence of randomly missing data. We showed that the MUSIC DOA estimate was consistent for a single source even in the presence of missing data. The analysis based on random matrix theory uncovered the role of sample singular vector estimation accuracy in the MSE performance of the MUSIC DOA estimate. Furthermore, it brought into sharp focus the phase transition phenomenon in the consistency of the estimate and its MSE performance.

Extension of this characterization to the multiple source setting is an important open problem. Future directions for work include the study of other missing data models, *e.g.*, in the case of sensor faults, or correlated missing data.

In Chapters III and IV, we found that exploiting the Kronecker product structure of the singular vector led to improved estimation performance. Furthermore, we also discovered via simulations, and random matrix theory based analysis, that first computing the SVD of the data and then exploiting the Kronecker product structure gave more accurate estimates than first exploiting the Kronecker product structure, and then using the SVD.

We used a multi-rank version of the KPCA2,1 algorithm to initialize the ALS algorithm in Chapter VI. A formal derivation of multi rank analogues to these algorithms will have several applications: in improved performance in STAP and MIMO STAP systems, in the analysis of tensor decomposition algorithms, and in numerical linear algebra computations involving tensors, to name a few.

In Chapter V, we derived an optimal weight for the estimation inverse covariance matrix for low rank signals. We characterized the asymptotic behavior of these optimum weights,

and proposed an algorithm called BeamShrink, which improved the performance of SVD based beamformers. The role of sample singular vector bias and sample singular vector estimation accuracy was highlighted in the shrinkage nature of the optimum weights. We also showed that BeamShrink was robust to rank overestimation. In Chapter VI, we proposed an algorithm called TensorShrink for improved denoising of low rank tensor estimates, and TensorBeamshrink for improved performance of tensor estimation based beamformers.

It is pleasantly surprising that new mathematical tools enable us to find analytical and data driven answers to optimization problems which involve unobservable quantities. We also recognized that the complete solution to these in Chapter VI depended on objects whose characterizations are open problems, but which are similar in spirit to simpler objects which have been fully characterized in recent random matrix theory literature.

It is exciting to look forward to new advances which will enable us to derive exact solutions to these open problems, and also cast a new light on already solved classical problems.

APPENDIX A

Pertinent Results from Random Matrix Theory

From the expression for the objective function in (2.6b), we note that characterizing the inner product $u(\theta)^H \widehat{u}$, which quantifies the accuracy of the sample singular vector relative to the true underlying singular vector, is critical to analyzing the performance of the MUSIC algorithm. In this section, we present a characterization of this inner product in the large array regime. This characterization utilizes, and extends, some recent results from random matrix theory.

Paul [36] characterized the estimation accuracy of the sample eigenvectors of the 'spiked covariance matrix' model. Benaych-Georges *et. al.* [13] provided a more general characterization of sample eigenvectors in a 'signal-plus-noise' setting. Onatski [40] and Benyach-Georges *et. al* [14] extended the analysis to rectangular matrices and their singular vectors. Nadakuditi [15] extended the characterization of singular vector accuracy to the missing data setting.

The analysis presented here extends the results of [14] to the missing data setting, drawing upon the approach in [15]. The almost sure limit characterization of $|\widehat{u}^H q|^2$, where q is a vector orthogonal to $u(\theta)$, is new and essential to the MUSIC consistency analysis, as used in Section II.3 for proving Proposition II.1 (see lines after (A.1)).

Theorem A.1. *Let $u(\theta) \in \mathbb{C}^{m\times 1}$, and $s \in \mathbb{C}^{n\times 1}$ be the largest left and right singular vectors of the matrix S in (2.3). Assume that $||u(\theta)||_\infty < \eta_u \frac{\log^{C_u}(m)}{\sqrt{m}}$, and $||s||_\infty < \eta_v \frac{\log^{C_v}(n)}{\sqrt{n}}$, for some $\eta_u, \eta_v, C_u, C_v > 0$, where $||.||_\infty$ denotes the supremum norm. Let $\widehat{u}$ be the left singular vector associated with the largest singular value of $\widehat{Y}$ as in (2.5). Let $q \in \mathbb{C}^{m\times 1}$ be an arbitrary, unit*

norm vector such that $q^H u(\theta) = 0$. *Then, as* $m, n \to \infty$ *and* $m/n \to c \in (0, \infty)$, *we have that*

$$|u(\theta)^H \widehat{u}|^2 \xrightarrow{\text{a.s.}} \begin{cases} \alpha^2 := 1 - \dfrac{c(1+p\sigma^2)}{p\sigma^2(p\sigma^2+c)} & \text{if } \sigma > \dfrac{c^{\frac{1}{4}}}{\sqrt{p}}, \\ 0 & \text{otherwise.} \end{cases} \tag{A.1a}$$

Moreover, when $\sigma > \dfrac{c^{1/4}}{\sqrt{p}}$, *then*

$$|\widehat{u}^H q|^2 \xrightarrow{\text{a.s.}} 0. \tag{A.1b}$$

Recall that the matrix $\widehat{Y}$ from (2.4), which may be written as

$$\widehat{Y} = Y \odot M = S \odot M + \underbrace{Z \odot M}_{=:\widetilde{Z}},$$

where $\odot$ denotes the Hadamard or element-wise product and M is the mask matrix given by

$$M_{ij} = \begin{cases} 1 & \text{with probability } p, \\ 0 & \text{with probability } 1-p. \end{cases} \tag{A.2}$$

For notational simplicity, we let

$$u := u(\theta).$$

Express $\widehat{Y}$ in (A) as $\widetilde{Y} + \Delta_{\widehat{Y}}$, thus:

$$\widehat{Y} = \underbrace{\mathbb{E}\left[S \odot M\right] + \widetilde{Z}}_{:=\widetilde{Y}} + \underbrace{\left(S \odot M - \mathbb{E}\left[S \odot M\right]\right)}_{=:\Delta_{\widehat{Y}}}. \tag{A.3}$$

Since $\mathbb{E}\left[S \odot M\right] = pS$, we get:

$$\widetilde{Y} = pS + \widetilde{Z} = p\sigma u s^H + \widetilde{Z}. \tag{A.4}$$

We shall prove Theorem A.1 as follows:

1. We first show that the spectral density of $\widetilde{Z}$ converges to a compactly supported, non random density in the asymptotic regime considered.
2. We then characterize the left singular vector $\widetilde{u}$ associated with its largest singular value of $\widetilde{Y}$ and show that it obeys the expressions in Theorem A.1.
3. We show that the operator norm of $\Delta_{\widehat{Y}}$ converges almost surely to zero in the large

matrix limit.

4. Finally, we conclude by showing that since $\widehat{Y} = \widetilde{Y} + \Delta_{\widehat{Y}}$, the fact that the operator norm of $\Delta_{\widehat{Y}}$ converges to zero almost surely implies the statements of Theorem A.1.

A.1 Limiting Spectrum of $\widetilde{Z}$

Assume without loss of generality that $m \leq n$. Let $\widetilde{\sigma}_1 \geq \widetilde{\sigma}_2 \geq \ldots \geq \widetilde{\sigma}_m > 0$ be the ordered singular values of $\widetilde{Z}$. Consider the measure

$$\mu_{\widetilde{Z}_n} = \frac{1}{m} \sum_{i=1}^{m} \delta_{\widetilde{\sigma}_i}, \tag{A.5}$$

which is the empirical distribution of the singular values of $\widetilde{Z}$, where $\delta_{\widetilde{\sigma}_i}$ is the Dirac delta function. Note that

$$\mathbb{E}\left[\widetilde{Z}_{ij}\right] = \mathbb{E}\left[\widetilde{Z}_{ij} \,\middle|\, M_{ij} = 0\right] + \mathbb{E}\left[\widetilde{Z}_{ij} \,\middle|\, M_{ij} = 1\right] = 0, \tag{A.6a}$$

$$\mathbb{E}\left[\widetilde{Z}_{ij}^2\right] = \mathbb{E}\left[\widetilde{Z}_{ij}^2 \,\middle|\, M_{ij} = 0\right] + \mathbb{E}\left[\widetilde{Z}_{ij}^2 \,\middle|\, M_{ij} = 1\right] = \frac{p}{m}. \tag{A.6b}$$

Since $\widetilde{Z} = Z \odot M$, and the elements of Z have bounded higher order moments, so will the elements of $\widetilde{Z}$. Therefore, the elements of $\widetilde{Z}$ are independent, and identically distributed with mean 0, variance p/m and bounded higher order moments. Then, from [100, Theorem 3.10, pg. 51], as $m, n \to \infty$, $m/n \to c \in (0, 1]$,

$$\mu_{\widetilde{Z}_n}(x) \xrightarrow{\text{a.s.}} \mu_{\widetilde{Z}}$$

where

$$d\mu_{\widetilde{Z}}(x) = \frac{\sqrt{(b - x^2)(x^2 - a)}}{\pi x p c} I_{[a,b]}(x), \tag{A.7}$$

where $a = \sqrt{p}(1 - \sqrt{c})$, $b = \sqrt{p}(1 + \sqrt{c})$, and $I_{[a,b]}(\cdot)$ is the indicator function. Moreover, from [100, Theorem 3.10, pg. 51] it follows that $\widetilde{\sigma}_1 \xrightarrow{\text{a.s.}} b = \sqrt{p}(1 + \sqrt{c})$.

A.2 Largest Singular Value and Corresponding Left Singular Vector of $\widetilde{Y}$

Let $\widetilde{u}$ be the left singular vector of $\widetilde{Y}$ defined in (A.4), corresponding to its largest singular value $\widetilde{\sigma}$. We will show that $|\widetilde{u}^H u|^2$ and $|\widetilde{u}^H q|^2$, (where q is any unit vector orthogonal to u) converge almost surely to the same limiting values as those of $|\widehat{u}^H u^2|$ and $|\widehat{u}^H q|^2$, given by (A.1a) and (A.1b) respectively. The approach leading to [14, Theorem 2.10] can be used to characterize $|\widetilde{u}^H u|^2$, but does not characterize $|\widetilde{u}^H q|^2$; here we take an alternative approach which allows us to characterize $|\widetilde{u}^H q|^2$ as well. We begin by deriving an expression for $\widetilde{u}$:

Proposition A.1. *For the matrix $\widetilde{Y}$ defined in (A.4),*

$$\widetilde{u} = \frac{\left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right) u}{\sqrt{u^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right)^2 u}}, \tag{A.8}$$

where

$$A_{\widetilde{Z}} = \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \tag{A.9a}$$

$$B_{\widetilde{Z}} = \frac{1}{d_{\widetilde{Z}}} \left(w^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w - \frac{1}{p\sigma} \right) \; u w^H \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \tag{A.9b}$$

$$C_{\widetilde{Z}} = \frac{1}{d_{\widetilde{Z}}} \; w^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) \; u u^H \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \tag{A.9c}$$

$$D_{\widetilde{Z}} = \frac{1}{d_{\widetilde{Z}}} \; u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) \; w w^H \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \tag{A.9d}$$

$$E_{\widetilde{Z}} = \frac{1}{d_{\widetilde{Z}}} \left(w^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u - \frac{1}{p\sigma} \right) \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) \; w u^H \; \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \tag{A.9e}$$

in which

$$\mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) = \left(\widetilde{\sigma}^2 I - \widetilde{Z}\widetilde{Z}^H \right)^{-1}, \tag{A.9f}$$

$$d_{\widetilde{Z}} = \det \begin{bmatrix} w^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u - \dfrac{1}{p\sigma} & w^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w \\ u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u & u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w - \dfrac{1}{p\sigma} \end{bmatrix},$$

and

$$w = \widetilde{Z}^H s \tag{A.9g}$$

where s is defined in (2.3).

Proof. Given some Hermetian matrix X, let

$$\widetilde{X} = X + \lambda_X u_X u_X^H,$$

$\widetilde{\lambda}_X$ be the largest eigenvalue of $\widetilde{X}$, and $\widetilde{u}_X$ the associated eigenvector. Then, $\widetilde{\lambda}_X, \widetilde{u}_X$ satisfy the eigenvalue-eigenvector relationship:

$$\begin{aligned} & (X + \lambda_X u_X u_X^H)\,\widetilde{u}_X = \widetilde{\lambda}_X\,\widetilde{u}_X, \\ \text{so that} \quad & \widetilde{u}_X = \left(\widetilde{\lambda}_X I - X\right)^{-1} \lambda_X\,(u_X^T \widetilde{u}_X)\,u_X \\ \text{which implies} \quad & \widetilde{u} \propto \left(\widetilde{\lambda}_X I - X\right)^{-1} u_X. \end{aligned}$$

Since $\widetilde{u}_X$ is along the direction of $\left(\widetilde{\lambda}_X I - X\right)^{-1} u_X$, we may determine $\widetilde{u}_X$ by finding the unit norm vector along that direction:

$$\widetilde{u}_X = \frac{\left(\widetilde{\lambda}_X I - X\right)^{-1} u_X}{\sqrt{u_X^H \left(\widetilde{\lambda}_X I - X\right)^{-2} u_X}}. \tag{A.10}$$

We are interested in $\widetilde{u}$, the eigenvector of $\widetilde{Y}\widetilde{Y}^H$ corresponding to its largest eigenvalue, $\widetilde{\sigma}^2$. From (A.4), and using (A.9g),

$$\widetilde{Y}\widetilde{Y}^H = \widetilde{Z}\widetilde{Z}^H + p\sigma\; uw^H + p\sigma\; wu^H + p^2\sigma^2\,(s^H s)\,uu^H.$$

Substituting

$$u_X = u,\;\; \widetilde{\lambda}_X = \widetilde{\sigma}_1^2,\;\; \widetilde{X} = \widetilde{Y}\widetilde{Y}^H,\;\; X = \widetilde{Z}\widetilde{Z}^H + p\sigma u w^H + p\sigma w u^H,$$

into (A.10), and rewriting the expression in matrix-vector form gives us

$$\widetilde{u} = \frac{\left(\widetilde{\sigma}^2 I - \widetilde{Z}\widetilde{Z}^H - \begin{bmatrix} u & w \end{bmatrix} \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix} \begin{bmatrix} w & u \end{bmatrix}\right)^{-1} u}{\sqrt{u^H \left(\widetilde{\sigma}^2 I - \widetilde{Z}\widetilde{Z}^H - \begin{bmatrix} u & w \end{bmatrix} \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix} \begin{bmatrix} w & u \end{bmatrix}\right)^{-2} u}}. \tag{A.11}$$

To simplify the expression further, we shall use the matrix inversion lemma [18, Sec. 6.4, Pg. 48], which states that, given matrices W_1, W_2, W_3, W_4 of appropriate dimensions:

$$(W_1 + W_2 W_3 W_4)^{-1} = W_1^{-1} - W_1^{-1} W_2 \left(W_3^{-1} + W_4 W_1^{-1} W_2\right)^{-1} W_3 W_1^{-1}. \tag{A.12}$$

In (A.12), substitute:

$$W_1 = \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}), \quad W_2 = \begin{bmatrix} u & w \end{bmatrix}, \quad W_3 = \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix}, \quad W_4 = \begin{bmatrix} w & u \end{bmatrix}$$

to obtain

$$\left(\widetilde{\sigma}^2 I - \widetilde{Z}\widetilde{Z}^H - \begin{bmatrix} u & w \end{bmatrix} \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix} \begin{bmatrix} w & u \end{bmatrix}\right)^{-1} = A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}} \tag{A.13}$$

where $A_{\widetilde{Z}}, \ldots, E_{\widetilde{Z}}$ are defined in (A.9). Substituting (A.13) in (A.11) gives us the expression for $\widetilde{u}$ in (A.8). □

Using the expression for $\widetilde{u}$ given by (A.8) from Proposition A.1, we can find the quantities of interest, $|u^H \widetilde{u}|^2$, and $|q^H \widetilde{u}|^2$:

$$|u^H \widetilde{u}|^2 = \frac{\left(u^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right) u\right)^2}{u^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right)^2 u}, \tag{A.14a}$$

$$|q^H \widetilde{u}|^2 = \frac{\left(q^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right) u\right)^2}{u^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right)^2 u}. \tag{A.14b}$$

If we evaluate (A.14) by plugging in the definition of $\mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma})$, we see that $|u^H \widetilde{u}|^2$, and $|q^H \widetilde{u}|^2$ are dependent on bilinear forms involving the resolvent of $\widetilde{Z}\widetilde{Z}^H$, e.g. $u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w$, $u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u$ and so on. Therefore, we can find the limits of $|u^H \widetilde{u}|^2$ and $|q^H \widetilde{u}|^2$ by finding the limits of the aforementioned bilinear forms. We will show that the expressions on the right hand side of (A.14) will converge almost surely to the right hand side of the corresponding terms given by (A.1). Specifically, by [41, Theorems 2.4 and 2.5], and using the fact derived in Appendix A.1 that $\mu_{\widetilde{Z}_n} \xrightarrow{\text{a.s.}} \mu_{\widetilde{Z}}$, we have that

$$u^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w = u^H \left(z^2 I - \widetilde{Z}\widetilde{Z}^H\right)^{-1} \widetilde{Z} s \xrightarrow{\text{a.s.}} 0, \tag{A.15a}$$

$$q^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) u \;= q^H \left(z^2 I - \widetilde{Z}\widetilde{Z}^H\right)^{-1} u \;\xrightarrow{\text{a.s.}}\; 0, \tag{A.15b}$$

$$q^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma}) w = q^H \left(z^2 I - \widetilde{Z}\widetilde{Z}^H\right)^{-1} \widetilde{Z} s \xrightarrow{\text{a.s.}} 0. \tag{A.15c}$$

Using (A.15) in (A.9), we see that

$$q^H \left(A_{\widetilde{Z}} - B_{\widetilde{Z}} + C_{\widetilde{Z}} + D_{\widetilde{Z}} - E_{\widetilde{Z}}\right) u \xrightarrow{\text{a.s.}} 0$$

so that the numerator of $|q^H \widetilde{u}|^2$ in (A.14) converges almost surely to 0. To show that $|\widetilde{u}^H q|^2 \xrightarrow{\text{a.s.}} 0$, we need to show that the denominator of $|\widetilde{u}^H q|^2$ in (A.14) does not also converge to 0. We can show this, and complete the characterization of $|u^H \widetilde{u}|^2$, by using results from [14].

The limit of $|u^H \widetilde{u}|^2$ is obtained by application of [14, Theorem 2.10] to (2.3) (by setting, in the notation of that paper,

$$\theta = p\sigma, \;\; \widetilde{X} = \widetilde{Y}, \;\; X = \widetilde{Z}.$$

Doing so, we get that as $m, n \to \infty, m/n \to c$:

$$|u^H \widetilde{u}|^2 \xrightarrow{\text{a.s.}} \frac{-2\, \varphi^2_{\mu_{\widetilde{Z}}}(\rho)}{p^2\, \sigma^2\, D^{(1)}_{\mu_{\widetilde{Z}}}(\rho)} \tag{A.16}$$

where $D^{(1)}_{\mu_{\widetilde{Z}}}(\rho)$ is the first derivative of the so called D-transform of the measure $\mu_{\widetilde{Z}}$, defined for $x > b$ as

$$D_{\mu_{\widetilde{Z}}}(x) := \left[\int \frac{x}{x^2 - t^2} d\mu_{\widetilde{Z}}(t)\right] \left[c \int \frac{x}{x^2 - t^2} d\mu_{\widetilde{Z}}(t) + \frac{1 - x}{c}\right], \tag{A.17}$$

$$\varphi_{\mu_{\widetilde{Z}}}(x) := \int \frac{x}{x^2 - t^2} d\mu_{\widetilde{Z}}(t), \;\text{ for } x > b, \tag{A.18}$$

b being the edge of the spectrum of $\widetilde{Z}$ as in (A.7). Here ρ is the limit of $\widetilde{\sigma}$ in the large array limit considered, and is given by [14, Theorem 2.9]:

$$\widetilde{\sigma} \xrightarrow{\text{a.s.}} \rho := \begin{cases} D^{-1}_{\mu_{\widetilde{Z}}}\left(\dfrac{1}{p\sigma^2}\right) & \text{if } p\,\sigma > \left(D_{\mu_{\widetilde{Z}}}(b^+)\right)^{-1/2} = \sqrt{p} c^{1/4} \\ b & \text{otherwise} \end{cases}, \tag{A.19}$$

where $D^{-1}_{\mu_{\widetilde{Z}}}(\cdot)$ is the functional inverse of the D-transform defined in (A.17) and $D_{\mu_{\widetilde{Z}}}(b^+) := \lim_{x \downarrow b} D_{\mu_{\widetilde{Z}}}(x)$. The explicit expression of $|u^H \widetilde{u}|^2$ is obtained by computing the transforms in

(A.16) for the limiting spectral distribution of $\widetilde{Z}$ given by (A.7), as in [14, (10)]. Doing so, we find that, as $m, n \to \infty, m/n \to c$:

$$|u^H \widetilde{u}|^2 \xrightarrow{\text{a.s.}} \begin{cases} 1 - \dfrac{c(1+p\sigma^2)}{p\sigma^2(p\sigma^2+c)} & \text{if } \sqrt{p}\,\sigma > c^{1/4} \\ 0 & \text{otherwise} \end{cases}. \tag{A.20}$$

Now we complete the characterization of $|q^H \widetilde{u}|^2$. From (A.20), if $\sigma > c^{1/4}/\sqrt{p}$, $|u^H \widetilde{u}|^2$ converges almost surely to a finite, deterministic value in the regime of our interest. Furthermore, from (A.14), we note that the denominators of both$|u^H \widetilde{u}|^2$ and $|q^H \widetilde{u}|^2$ are the same. When $\sigma > c^{1/4}/\sqrt{p}$, then the denominator of $|q^H \widetilde{u}|^2$ in (A.14) converges almost surely to a non-zero value. This, coupled with the argument following (A.15) that the numerator of $|q^H \widetilde{u}|^2$ converges almost surely to 0, implies that

$$|q^H \widetilde{u}|^2 \xrightarrow{\text{a.s.}} 0 \tag{A.21}$$

where q is a unit vector orthogonal to u.

Equation (A.19) uncovers the root of the phase transition phenomenon in the performance of the MUSIC algorithm. The result states that, as long as the product of the probability of observing entries in the data and the SNR is above a certain threshold, given by $\left(D_{\mu_{\widetilde{Z}}}(b^+)\right)^{-1/2}$, $\widetilde{\sigma}_1$ separates from the spectrum of the noise singular values, $\mu_{\widetilde{Z}}$. If either p or σ is low enough so that their product drops below the critical threshold, the $\widetilde{\sigma}$ sits at the edge of the noise spectrum, so that signal is indistinguishable from noise. For MSE characterization of MUSIC in this work, we assume that we are operating in the regime where $p\sigma > \left(D_{\mu_{\widetilde{Z}}}(b^+)\right)^{-1/2}$.

A.3 Largest singular value of $\Delta_{\widehat{Y}}$

In this section, we will characterize the behavior of largest singular value of the matrix $\Delta_{\widehat{Y}}$ defined in (A.3). Since $\Delta_{\widehat{Y}}$ is not independent of the mean data matrix $\mathbb{E}[S]$, we cannot repeat the 'signal-plus-noise-matrix' technique used to characterize the singular value and vectors of $\widetilde{Y}$ as in the previous section. Instead, our proof strategy is to show that largest singular value of $\Delta_{\widehat{Y}}$, denoted $\sigma_1(\Delta_{\widehat{Y}})$, converges almost surely to 0 and to use this fact to complete the characterization of the leading left singular vector of the matrix $\widehat{Y}$.

Theorem A.2. *Assuming the hypothesis of Theorem A.1 is satisfied and $\Delta_{\widehat{Y}}$ is defined as*

in (A.3). *Then,*

$$\sigma_1(\Delta_{\widehat{Y}}) \xrightarrow{\text{a.s.}} 0, \tag{A.22}$$

as $m, n \to \infty, m/n \to c \in (0, \infty)$.

Proof. The $(i,j)^{\text{th}}$ elements of $\Delta_{\widehat{Y}} = S \odot M - pS$, defined in (A.3), are independent since the elements of M are i.i.d., and are given by

$$\left(\Delta_{\widehat{Y}}\right)_{ij} = \sigma\, u_i\, s_j\, M_{ij} - p\, \sigma\, u_i\, s_j. \tag{A.23}$$

Consequently,

$$\mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}\right] = pS_{ij} - pS_{ij} = 0$$

and

$$\mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}^2\right] = p(1-p)\, S_{ij}^2, \tag{A.24a}$$

$$\mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}^4\right] = \left(-3p^4 + 6p^3 - 4p^2 + p\right)\, S_{ij}^4. \tag{A.24b}$$

The low coherence assumption on the elements of u and v in the hypothesis of Theorem A.1, implies that when $m = O(n)$, we have that

$$\max_{i,j} |S_{ij}| \leq \|u\|_\infty \|s\|_\infty = O\left(\frac{\log\, m \text{ factors}}{m}\right). \tag{A.25}$$

Equations (A.25), (A.24a) and (A.24b) imply that the elements of $\Delta_{\widehat{Y}}$ have bounded fourth moment. Moreover, since the elements of $\Delta_{\widehat{Y}}$ are independent, zero mean random variables, with bounded fourth moment, we can directly apply Latala's theorem [101] to characterize the largest singular value of $\Delta_{\widehat{Y}}$. We have that:

$$\mathbb{E}\left[\sigma_1(\Delta_{\widehat{Y}})\right] \leq C\left[\max_i \left(\sum_j \mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}^2\right]\right)^{1/2} + \max_j \left(\sum_i \mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}^2\right]\right)^{1/2} + \left(\sum_{i,j} \mathbb{E}\left[(\Delta_{\widehat{Y}})_{ij}^4\right]\right)^{1/4}\right] \tag{A.26}$$

where C is a universal constant that does not depend on m, n. Plugging in (A.24a) and (A.24b) in (A.26), gives us

$$\mathbb{E}\left[\sigma_1(\Delta_{\widehat{Y}})\right] \leq O\left(\frac{\log m \text{ factors}}{\sqrt{m}}\right),$$

which implies that as $m, n \to \infty$,

$$\mathbb{E}\left[\sigma_1(\Delta_{\widehat{Y}})\right] \to 0.$$

We now establish a tail inequality on the largest singular value so we may show that $\sigma_1\left(\Delta_{\widehat{Y}}\right) \xrightarrow{\text{a.s.}} 0$. To that end, we first note that by Weyl's interlacing inequality [102, Sec. 7, Pg. 423], we have

$$|\sigma_1(\Delta_{\widehat{Y}} + \widetilde{Y}) - \sigma_1(\widetilde{Y})| \leq 1 \cdot \sigma_1\left(\Delta_{\widehat{Y}}\right). \tag{A.27}$$

Eq. (A.27) implies that the largest singular value is 1-Lipschitz continuous function of the entries of the matrix. Furthermore, $\sigma_1(\cdot)$ is a convex function of the entries of the matrix, since for any matrices W_1, W_2 and $\epsilon \in [0,1]$,

$$\sigma_1\left(\epsilon\, W_1 + (1-\epsilon)\, W_2\right) \leq \epsilon\, \sigma_1\left(W_1\right) + (1-\epsilon)\, \sigma_1\left(W_2\right).$$

Finally, from (A.25) we have that the elements of $\Delta_{\widehat{Y}}$ are bounded so that

$$\max_{ij} |\left(\Delta_{\widehat{Y}}\right)| \leq \mathcal{O}\left(\frac{\log\, m \text{ factors}}{m}\right) := \mathcal{K}.$$

Since the elements of $\Delta_{\widehat{Y}}$ are independent and bounded and $\sigma_1(\cdot)$ is a 1-Lipschitz, convex function, we may apply Talagrand's inequality [103, Theorem 2.1.13], to get the tail bound

$$\mathbb{P}\left(\left|\sigma_1\left(\Delta_{\widehat{Y}}\right) - \mathbb{E}\left[\sigma_1\left(\Delta_{\widehat{Y}}\right)\right]\right| \geq \epsilon\right) \leq 2\, \exp\left(-\frac{\bar{C}\epsilon^2}{\mathcal{K}^2}\right) = 2\exp\left(-\frac{\bar{C}\epsilon^2 m^2}{\log\, m \text{ factors}}\right) \tag{A.28}$$

where $\bar{C}$ is a universal constant that does not depend on m, n. Note that

$$\sum_m 2\exp\left(-\frac{\bar{C}\epsilon^2 m^2}{\log\, m \text{ factors}}\right) < \infty.$$

Hence, via the Borel-Cantelli Lemma, we can conclude that

$$\sigma_1(\Delta_{\widehat{Y}}) \xrightarrow{\text{a.s.}} 0.$$

□

A.4 Relating the Singular Values and Vectors of $\Delta_{\widehat{Y}}$ and $\widetilde{Y}$

In Section A.2, we characterized the limiting behaviour of $|\widehat{u}^H u|^2$, where $\widetilde{u}$ was the largest left singular vector of $\widetilde{Y}$, as in (A.4). In this section, we will use the results of Section A.3

to complete the proof of Theorem A.1, by showing that $|\widehat{u}^H u|^2$ converges to the same limit as $|\widetilde{u}^H u|^2$ in the asymptotic regime considered, where $\widehat{u}$ is the largest left singular vector of $\widehat{Y}$, as in (A), corresponding to its largest singular value $\widehat{\sigma}$.

As in Section A.2, we first find $\widehat{u}$, which is the eigenvector of $\widehat{Y}\widehat{Y}^H$, corresponding to its largest eigenvalue, $\widehat{\sigma}^2$. This may be evaluated by substituting

$$u_X = u, \;\; \widetilde{\lambda}_X = \widehat{\sigma}^2, \;\; \widetilde{X} = \widehat{Y}\widehat{Y}^H, \;\; X = (\widetilde{Z} + \Delta_{\widehat{Y}})(\widetilde{Z} + \Delta_{\widehat{Y}})^H + p\sigma u w^H + p\sigma w u^H$$

in (A.10). We have that:

$$\widehat{u} = \frac{\left(\widehat{\sigma}^2 I - \left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)^H - \begin{bmatrix} u & w \end{bmatrix} \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix} \begin{bmatrix} w & u \end{bmatrix}\right)^{-1} u}{\sqrt{u^H \left(\widehat{\sigma}^2 I - \left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)^H - \begin{bmatrix} u & w \end{bmatrix} \begin{bmatrix} -p\sigma & 0 \\ 0 & -p\sigma \end{bmatrix} \begin{bmatrix} w & u \end{bmatrix}\right)^{-2} u}}. \tag{A.29}$$

Next, we obtain expressions for $|\widehat{u}^H u|^2$ and $|\widehat{u}^H q|^2$. We follow the steps from (A.12) through to (A.14), but at each step we replace $\left(\widetilde{\sigma}_{\widetilde{Y}}^2 I - \widetilde{Z}\widetilde{Z}^H\right)$ by $\left(\widehat{\sigma}^2 I - \left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\right)$, or equivalently, replace $\mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma})$ with

$$\mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) := \left(\widehat{\sigma}^2 I - \left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\left(\widetilde{Z} + \Delta_{\widehat{Y}}\right)\right)^{-1}, \tag{A.30}$$

which is the term that differs between (A.11) and (A.29). This gives us

$$|\widehat{u}^H u|^2 = \frac{\left(u^H \left(A_{\widetilde{Z}+\Delta} - B_{\widetilde{Z}+\Delta} + C_{\widetilde{Z}+\Delta} + D_{\widetilde{Z}+\Delta} - E_{\widetilde{Z}+\Delta}\right) u\right)^2}{u^H \left(A_{\widetilde{Z}+\Delta} - B_{\widetilde{Z}+\Delta} + C_{\widetilde{Z}+\Delta} + D_{\widetilde{Z}+\Delta} - E_{\widetilde{Z}+\Delta}\right)^2 u}, \tag{A.31a}$$

$$|\widehat{u}^H q|^2 = \frac{\left(q^H \left(A_{\widetilde{Z}+\Delta} - B_{\widetilde{Z}+\Delta} + C_{\widetilde{Z}+\Delta} + D_{\widetilde{Z}+\Delta} - E_{\widetilde{Z}+\Delta}\right) u\right)^2}{u^H \left(A_{\widetilde{Z}+\Delta} - B_{\widetilde{Z}+\Delta} + C_{\widetilde{Z}+\Delta} + D_{\widetilde{Z}+\Delta} - E_{\widetilde{Z}+\Delta}\right)^2 u}, \tag{A.31b}$$

where

$$A_{\widetilde{Z}+\Delta} = \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}), \tag{A.32a}$$

$$B_{\widetilde{Z}+\Delta} = \frac{1}{d_{\widetilde{Z}+\Delta}} \left(w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w - \frac{1}{p\sigma}\right) \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) u w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) \tag{A.32b}$$

$$C_{\widetilde{Z}+\Delta} = \frac{1}{d_{\widetilde{Z}+\Delta}}\ w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma})\ uu^H\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}), \tag{A.32c}$$

$$D_{\widetilde{Z}+\Delta} = \frac{1}{d_{\widetilde{Z}+\Delta}}\ u^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) u\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma})\ ww^H\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}), \tag{A.32d}$$

$$E_{\widetilde{Z}+\Delta} = \frac{1}{d_{\widetilde{Z}+\Delta}}\ \left(w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) u - \frac{1}{p\sigma}\right)\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma})\ wu^H\ \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) \tag{A.32e}$$

in which

$$d_{\widetilde{Z}+\Delta} = \det \begin{bmatrix} w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) u - \frac{1}{p\sigma} & w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w \\ u^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) u & u^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w - \frac{1}{p\sigma} \end{bmatrix},$$

w is as in (A.9g) and $\mathcal{M}_{\widehat{Z}}(\widehat{\sigma})$ is as in (A.30).

We see that $|\widehat{u}^H u|^2, |\widehat{u}^H q|^2$ depend on the bilinear forms in (A.32), e.g. $w^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w$, $u^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma}) w$, *etc.* The expressions in (A.31) and (A.14) are of the same form, but with $\mathcal{M}_{\widehat{Z}}(\widehat{\sigma})$ in place of $\mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma})$. The limiting behaviour of $|\widetilde{u}^H u|^2$, $wtu^H q|^2$ in the regime considered was a consequence of the convergence of bilinear forms in (A.9) to deterministic quantities. To complete the proof of Theorem A.1, we will now show that the bilinear forms in (A.32) converge to those in (A.9), and thus $|\widehat{u}^H u|^2, |\widehat{u}^H q|^2$ must converge to the same limit as $|\widetilde{u}^H u|^2$, $\widetilde{u}^H q|^2$.

Theorem A.3. *Let $\widetilde{Y}$ and $\Delta_{\widehat{Y}}$ be matrices as defined in (A). Let $\sigma_i(\cdot)$ denote the i^{th} largest singular value of the matrix in the argument. Let $\mathcal{M}_{\widetilde{Y}}(\cdot)$ be as in (A.9f) and $\mathcal{M}_{\widetilde{Y}+\Delta}(\cdot)$ as in (A.30). Let v and w be any unit norm vectors, and let $z \in \mathbb{C}$. Then, as $m, n \to \infty, m/n \to c \in (0, \infty)$,*

$$\sigma_i(\Delta_{\widehat{Y}} + \widetilde{Y}) \xrightarrow{\text{a.s.}} \sigma_i(\widetilde{Y}), \tag{A.33a}$$

$$\sigma_1(\Delta_{\widehat{Y}} + \widetilde{Y}) \xrightarrow{\text{a.s.}} \sigma_1(\widetilde{Y}), \tag{A.33b}$$

$$v^H \mathcal{M}_{\widetilde{Y}+\Delta}(z) w \xrightarrow{\text{a.s.}} v^H \mathcal{M}_{\widetilde{Y}}(z) w. \tag{A.33c}$$

Proof. Using Weyl's interlacing inequality [102, Sec. 7, Pg. 423], we have that

$$|\sigma_i(\Delta_{\widehat{Y}} + \widetilde{Y}) - \sigma_i(\widetilde{Y})| \leq \sigma_1(\Delta_{\widehat{Y}}) \tag{A.34}$$

for $i = 1, \ldots, \min(m, n)$. In particular, when $i = 1$ in (A.34), have that:

$$|\sigma_1(\Delta_{\widehat{Y}} + \widetilde{Y}) - \sigma_1(\widetilde{Y})| \leq \sigma_1(\Delta_{\widehat{Y}}).$$

In Theorem A.2, we showed that $\sigma_1(\Delta_{\widehat{Y}}) \xrightarrow{\text{a.s.}} 0$, which proves (A.33a) and (A.33b).

Given any two Hermetian matrices M_1, M_2, and their resolvents $\mathcal{M}_i(z) = (zI - M_i)^{-1}$, $i = 1, 2$, the resolvent identity [104, Prop 3.1 (iv)] states that, for $\Im z > 0$,

$$\mathcal{M}_2(z) - \mathcal{M}_1(z) = \mathcal{M}_2(z)(M_2 - M_1)\mathcal{M}_1(z) \tag{A.35}$$

where $\Im z$ denotes the imaginary part of z. This result be verified by left multiplying by $(zI - M_2)$ and right multiplying by $(zI - M_1)$. Substituting

$$M_1 = \widetilde{Y}\widetilde{Y}^H, \;\; M_2 = \left(\widetilde{Y} + \Delta_{\widehat{Y}}\right)\left(\widetilde{Y} + \Delta_{\widehat{Y}}\right)^H,$$

in (A.35), we get

$$\begin{aligned}\left(zI - (\widetilde{Y} + \Delta_{\widehat{Y}})(\widetilde{Y} + \Delta_{\widehat{Y}})^H\right)^{-1} - (zI - \widetilde{Y}\widetilde{Y}^H)^{-1} \\ = \left(zI - (\widetilde{Y} + \Delta_{\widehat{Y}})(\widetilde{Y} + \Delta_{\widehat{Y}})^H\right)^{-1}\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)(zI - \widetilde{Y}\widetilde{Y}^H).\end{aligned}$$

Rewriting this using the notation of (A.9f) and (A.30) we get

$$\mathcal{M}_{\widetilde{Y}+\Delta}(z) - \mathcal{M}_{\widetilde{Y}}(z) = \mathcal{M}_{\widetilde{Y}+\Delta}(z)\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)\mathcal{M}_{\widetilde{Y}}(z). \tag{A.36}$$

Left multiplying (A.36) by v^H and right multiplying by w gives us

$$v^H\left(\mathcal{M}_{\widetilde{Y}+\Delta}(z) - \mathcal{M}_{\widetilde{Y}}(z)\right)w = v^H\left(\mathcal{M}_{\widetilde{Y}+\Delta}(z)\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)\mathcal{M}_{\widetilde{Y}}(z)\right)w,$$

so that

$$\begin{aligned}\left|v^H\mathcal{M}_{\widetilde{Y}+\Delta}(z)w - v^H\mathcal{M}_{\widetilde{Y}}(z)w\right| &= \left|v^H\mathcal{M}_{\widetilde{Y}+\Delta}(z)\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)\mathcal{M}_{\widetilde{Y}}(z)w\right| \\ &\le \sigma_1\left(\mathcal{M}_{\widetilde{Y}+\Delta}(z)\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)\mathcal{M}_{\widetilde{Y}}(z)\right) \\ &\le \frac{\sigma_1\left(\Delta_{\widehat{Y}}\widetilde{Y}^H + \widetilde{Y}\Delta_{\widehat{Y}}^H + \Delta_{\widehat{Y}}\Delta_{\widehat{Y}}^H\right)}{(\Im z)^2},\end{aligned} \tag{A.37}$$

where the last inequality follows from [104, Prop 3.1 (iii)]. Using Weyl's inequalities [102, Theorem 3.3.16-(a),(d) Pg. 178] which state that

$$\sigma_1(A + B) \le \sigma_1(A) + \sigma_1(B) \quad \text{and} \quad \sigma_1(AB) \le \sigma_1(A) \cdot \sigma_1(B),$$

equation (A.37) becomes

$$|v^H \mathcal{M}_{\widetilde{Y}+\Delta}(z)w - v^H \mathcal{M}_{\widetilde{Y}}(z)w| \leq \frac{\sigma_1(\Delta_{\widehat{Y}})\left(\sigma_1(\widetilde{Y}^H) + \sigma_1(\widetilde{Y}) + \sigma_1(\Delta_{\widehat{Y}}^H)\right)}{(\Im z)^2}.$$

Since, by Theorem A.2, $\sigma_1(\Delta_{\widehat{Y}}) \xrightarrow{\text{a.s.}} 0$, we have completed the proof of (A.33c). □

We are now equipped to prove that the quantities in (A.32) converge to the same limit as those in (A.9). If $\mu_{\widetilde{Z}+\Delta_{\widehat{Y}_n}}$ is the empirical spectral density of $\widehat{Y}$ in (A), defined similarly to (A.5), then (A.33a) implies that, as $m, n \to \infty, m/n \to c$,

$$\mu_{\widetilde{Z}+\Delta_{\widehat{Y}_n}} \xrightarrow{\text{a.s.}} \mu_{\widetilde{Z}},$$

or that the limiting spectral densities of $\widehat{Y}$ and $\widetilde{Y}$ are equal. This means that any integral transform of $\mu_{\widetilde{Z}+\Delta_{\widehat{Y}_n}}$, such as (A.17) and (A.18) will be equal to the same transform of $\mu_{\widetilde{Z}}$. Combined with (A.33b), this implies that

$$\widehat{\sigma} \xrightarrow{\text{a.s.}} \rho,$$

where ρ is given by (A.19). We then use this result in (A.33c), and also set $\{v, w\}$ equal to the vectors $u(\theta)$ or the vector in (A.9g), to see that the bilinear forms such as $u(\theta)^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma})w$, $u(\theta)^H \mathcal{M}_{\widehat{Z}}(\widehat{\sigma})u$, converge to the corresponding bilinear forms $u(\theta)^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma})w$, $u(\theta)^H \mathcal{M}_{\widetilde{Z}}(\widetilde{\sigma})u$ and so on. Therefore, for the quantities defined in from (A.32) and (A.9),

$$A_{\widetilde{Z}+\Delta}, \ldots, E_{\widetilde{Z}+\Delta} \xrightarrow{\text{a.s.}} A_{\widetilde{Z}}, \ldots, E_{\widetilde{Z}} \tag{A.38}$$

respectively.

Using (A.38) in (A.31) and (A.14), we see that as $m/n \to \infty, m/n \to c$,

$$|\widehat{u}^H u|^2 \xrightarrow{\text{a.s.}} |\widetilde{u}^H u|^2,$$
$$|\widehat{u}^H q|^2 \xrightarrow{\text{a.s.}} |\widehat{u}^H q|^2,$$

whose limiting values were computed explicitly in (A.20). This completes the proof of Theorem A.1.

A.5 Justification for Conjecture II.1

Let

$$\bar{U} = \begin{bmatrix} u(\theta) & q & \bar{u}_3 & \dots & \bar{u}_m \end{bmatrix}$$

be a basis for the ambient space, where $u(\theta)$ is defined as in (2.1), and q is defined in (2.21a). Let $\widehat{u}$ be given by (2.5). Then, since $\widehat{u}$ is a unit norm vector,

$$||\bar{U}^H \widehat{u}||_2^2 = 1,$$

so that

$$|\widehat{u}^H u(\theta)|^2 + |\widehat{u}^H q|^2 + \sum_{k=3}^{n} |\widehat{u}^H \bar{u}_k|^2 = 1,$$

Therefore,

$$\mathbb{E}\left[|\widehat{u}^H q|^2\right] + \underbrace{\left(|\widehat{u}^H q|^2 - \mathbb{E}\left[|\widehat{u}^H q|^2\right]\right)}_{:=\Delta_{u_2}} + \sum_{k=3}^{m} \mathbb{E}\left[|\widehat{u}^H \bar{u}_k|^2\right] + \sum_{k=3}^{m} \underbrace{\left(|\widehat{u}^H \bar{u}_k|^2 - \mathbb{E}\left[|\widehat{u}^H \bar{u}_k|^2\right]\right)}_{:=\Delta_{u_k}}$$
$$= 1 - |\widehat{u}^H u(\theta)|^2$$

so that

$$\mathbb{E}\left[|\widehat{u}^H q|^2\right] + \sum_{k=3}^{m} \mathbb{E}\left[|\widehat{u}^H \bar{u}_k|^2\right] = 1 - |\widehat{u}^H u(\theta)|^2 - \sum_{k=2}^{m} \Delta_{u_k}. \tag{A.39}$$

Since, from Theorem A.1, $|\widehat{u}^H u(\theta)|^2 \xrightarrow{\text{a.s.}} \alpha^2$, we have that

$$\mathbb{E}\left[|\widehat{u}^H q|^2\right] + \sum_{k=3}^{m} \mathbb{E}\left[|\widehat{u}^H \bar{u}_k|^2\right] \xrightarrow{\text{a.s.}} 1 - \alpha^2 - \sum_{k=2}^{m} \Delta_{u_k}.$$

The uniform linear array is symmetric, since $|u(\theta)_k|^2 = 1/\sqrt{m}$ for $k = 1, \dots, m$ in (2.2). This motivates the conjecture that for $k = 2, \dots, m$,

$$\mathbb{E}\left[|\widehat{u}^H q|^2\right] = \mathbb{E}\left[|\widehat{u}^H \bar{u}_k|^2\right].$$

Assuming that this holds, we get

$$m\,\mathbb{E}\left[|\widehat{u}^H q|^2\right] \xrightarrow{\text{a.s.}} 1 - \alpha^2 - \sum_{k=2}^{m} \Delta_{u_k}.$$

Proving that $\sum_{k=2}^{m} \Delta_{u_k} \xrightarrow{\text{a.s.}} 0$ involves the characterization of the second order behavior of bilinear forms involving non-orthogonally invariant random matrices, which appear in e.g. (A.32). In the presence of missing data, the techniques required to characterize these bilinear forms go beyond the tools currently available in the literature, e.g. [105]. We leave this for future work. Here we conjecture that $\sum_{k=2}^{m} \Delta_{u_k} \xrightarrow{\text{a.s.}} 0$, which gives

$$m\,\mathbb{E}\left[|\widehat{u}^H q|^2\right] \xrightarrow{\text{a.s.}} 1-\alpha^2.$$

or

$$2m\,\mathbb{E}\left[\Re\left\{\langle u^\perp, \widehat{u}\rangle\right\}^2\right] \xrightarrow{\text{a.s.}} 1-\alpha^2, \tag{A.40}$$

where once again, by the symmetry of the uniform linear array, we conjecture that $\mathbb{E}\left[\Re|\widehat{u}^H q|^2\right] = \mathbb{E}\left[\Im|\widehat{u}^H q|^2\right]$, where $\Re\{\cdot\}$ and $\Im\{\cdot\}$ stand for 'real part of', and 'imaginary part of' respectively.

APPENDIX B

Manifold Vector Scaling for Uniform Linear Arrays

In this section, we will compute some of the terms involving the manifold vector (2.2) and its derivatives, which are used in the derivation of the MSE expression of MUSIC. The i^{th} element of the first derivative of the manifold vector evaluated at θ is

$$\left[u^{(1)}(\theta)\right]_i = \frac{1}{\sqrt{m}}\, e^{j\,2\,\pi\,d\,(i-1)\,\cos(\theta)}\,(-j\,2\,\pi\,d\,(i-1)\,\sin(\theta)). \tag{B.1}$$

Using (2.2) and (B.1),

$$\gamma_1 := u(\theta)^H u^{(1)}(\theta) = \sum_{k=0}^{m-1} -\frac{j\,2\,\pi\,d\,\sin(\theta)\,k}{m} = \mathcal{O}\,(m)\,. \tag{B.2}$$

Similarly,

$$\left\|u^{(1)}(\theta)\right\|_2^2 = \sum_{k=1}^{m} \frac{4\,\pi^2\,d^2\,\sin^2(\theta)\,k^2}{m} = \mathcal{O}\left(m^2\right). \tag{B.3}$$

From (B.2) and (B.3),

$$\delta_1^2 := \left\|u^{(1)}(\theta)\right\|_2^2 - |\gamma_1|^2 = \mathcal{O}\left(m^2\right). \tag{B.4}$$

Similarly, the i^{th} element of the second derivative of $u(\theta)$ is given by

$$\left[u^{(2)}(\theta)\right]_i = \frac{1}{\sqrt{m}} e^{j\,2\,\pi\,(i-1)\,d\,\cos(\theta)} \left(4\,\pi^2\,d^2\,\sin^2(\theta) - 2j\,\pi\,d\,\cos(\theta)\right), \tag{B.5}$$

so that

$$\gamma_2 := u(\theta)^H u^{(2)}(\theta) = \sum_{k=0}^{m-1} -\frac{4\,\pi^2\,d^2\,\sin^2(\theta) - 2j\,\pi\,d\,\cos(\theta)}{m}\,k^2 = \mathcal{O}\left(m^2\right). \tag{B.6}$$

Therefore,

$$||u^{(2)}(\theta)||_2^2 = \sum_{k=1}^{m} \frac{\left(16\,\pi^4\,d^4\,\sin^4(\theta) + 4\,\pi^2\,d^2\cos^2(\theta)\right)\,k^4}{m} = \mathcal{O}\left(m^4\right), \tag{B.7}$$

and

$$\delta_2^2 := \left\|u^{(2)}(\theta)\right\|_2^2 - |\gamma_2|^2 = \mathcal{O}\left(m^4\right). \tag{B.8}$$

APPENDIX C

Proof of Claim III.1

Let us consider each algorithm separately.

C.1 Analysis of KPCA2,1

From (3.44), for m large enough, we can use the approximation (3.46):

$$\widehat{u}_{\mathsf{svd}} = \alpha_{\mathsf{svd}}\, u + \sqrt{\frac{1-\alpha_{\mathsf{svd}}^2}{m}}\, g$$

where $g \sim \mathcal{CN}(0, I_m)$. Applying rearrangement $\mathcal{R}_{m_1\, m_2\times 1\to m_1\times m_2}(\widehat{u}_{\mathsf{svd}})$ on both sides,

$$\mathcal{R}(\widehat{u}_{\mathsf{svd}}) = \alpha_{\mathsf{svd}}\, ab^H + \sqrt{\frac{1-\alpha_{\mathsf{svd}}^2}{m}}\mathcal{R}(g). \tag{C.1}$$

The right hand side of (C.1) has a rank one signal plus noise structure as in (3.1), but unlike (3.1), it doesn't have identity covariance. Rescaling (C.1) so that it does, we get:

$$\sqrt{\frac{m_1}{1-\alpha_{\mathsf{svd}}^2}}\,\mathcal{R}(\widehat{u}_{\mathsf{svd}}) = \underbrace{\sqrt{\frac{\alpha_{\mathsf{svd}}\, m_1}{1-\alpha_{\mathsf{svd}}^2}}}_{:=\sigma_{\mathsf{kpca},1}}\, ab^H + \sqrt{\frac{1}{m_2}}\,\mathcal{R}(g). \tag{C.2}$$

The principal left and right singular vectors of the matrix on the left hand side of (C.2) are $\widehat{a}_{\mathsf{kpca2,1}}$ and $\widehat{b}_{\mathsf{kpca2,1}}$. By direct application of Proposition III.2, we have that:

$$|\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2 = \alpha^2\left(\sigma_{\mathsf{kpca},1},\, m_1/m_2\right), \tag{C.3a}$$

$$|\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2 = \beta^2 \left(\sigma_{\mathsf{kpca,1}},\, m_1/m_2\right). \tag{C.3b}$$

Since $\alpha^2_{\mathsf{svd}} \in [0,1] = \mathcal{O}(1)$, $\sigma_{\mathsf{kpca,1}} = \mathcal{O}(\sqrt{m_1})$ in (C.2). In our regime $m_1 = \mathcal{O}(m_2)$, $m_1/m_2 = \mathcal{O}(1)$. Using this in (C.3),

$$|\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2 \xrightarrow{\text{a.s.}} 1, \tag{C.4a}$$

$$|\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2 \xrightarrow{\text{a.s.}} 1. \tag{C.4b}$$

Since, from (3.7),

$$|\langle \widehat{u}_{\mathsf{kpca2,1}}, u\rangle|^2 = |\langle \widehat{a}_{\mathsf{kpca2,1}}, a\rangle|^2\, |\langle \widehat{b}_{\mathsf{kpca2,1}}, b\rangle|^2$$

we obtain (3.47a) as required.

We will now study the breakdown point of this algorithm. From (3.40a), the critical SNR above which $\widehat{a}_{\mathsf{kpca2,1}}$ and $\widehat{b}_{\mathsf{kpca2,1}}$ are correlated with the true underlying vectors satisfies

$$\sigma_{\mathsf{kpca,1}} = (m_1/m_2)^{1/4}.$$

Substituting $\sigma_{\mathsf{kpca,1}}$ from (C.2),

$$\frac{\alpha^2_{\mathsf{svd}}}{1-\alpha^2_{\mathsf{svd}}} = 1/\sqrt{m_1 m_2} = 1/\sqrt{m}. \tag{C.5}$$

Substituting for α^2_{svd} from (3.43) into (C.5), we obtain the equation

$$\frac{\sigma_0^4 - c_{\mathsf{svd}}}{c_{\mathsf{svd}}(1+\sigma_0^2)} = \frac{1}{\sqrt{m}} \implies \sigma_0^4 - \frac{c_{\mathsf{svd}}}{\sqrt{m}}\sigma_0^2 - \left(1+\frac{1}{\sqrt{m}}\right) c_{\mathsf{svd}} = 0. \tag{C.6}$$

Solving (C.6), and discarding the negative root since $\sigma_0 \in \mathbb{R}_+$, we get

$$\sigma^2_{\mathsf{crit,kpca2,1}} = \frac{\frac{c_{\mathsf{svd}}}{\sqrt{m}} + \sqrt{\frac{c^2_{\mathsf{svd}}}{m} + 4\left(1+\frac{1}{\sqrt{m}}\right) c_{\mathsf{svd}}}}{2} \tag{C.7}$$

In the regime of our interest, $m_1, m_2 \to \infty \implies m \to \infty$, we get

$$\sigma_{\mathsf{crit,kpca2,1}} = c_{\mathsf{svd}}^{1/4}$$

which is the *same* critical point around which there is a phase transition in SVD estimation performance.

C.2 Analysis of KPCA2,2ℓ

The rearrangement $\mathcal{R}_{m_1\, m_2\times n\to m_1\times m_2\, n}(Y)$ in Step 1 gives us

$$\mathcal{R}(Y) = \sigma_0\, a\, \mathrm{vec}(bs^H)^T + \mathcal{R}(Z) \tag{C.8}$$

which has a rank one signal plus noise structure. As we did in (C.2), we rescale the above equation so that the noise matrix has identity covariance:

$$\sqrt{\frac{1}{m_2}}\,\mathcal{R}(X) = \frac{\sigma_0}{\sqrt{m_2}} a\, \mathrm{vec}(bs^H)^T + \sqrt{\frac{1}{m_2}}\mathcal{R}(Z). \tag{C.9}$$

By direct application of Proposition III.2 to (C.9),

$$|\langle \widehat{a}_{\mathsf{kpca2,2}\ell}, a\rangle|^2 = \alpha^2\left(\frac{\sigma_0}{\sqrt{m_2}}, \frac{m_1}{m_2\, n}\right) = \frac{\sigma_0^4 - \dfrac{m_1\, m_2}{n}}{\sigma_0^2\left(\sigma_0^2 + \dfrac{m_2}{n}\right)} \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_p)} \tag{C.10a}$$

and

$$|\langle \widehat{v}_1, v_1\rangle|^2 = \beta^2_{\widehat{v},2\ell} = \frac{\sigma_0^4 - \dfrac{m_1\, m_2}{n}}{\sigma_0^2(\sigma_0^2 + m_2)} = O(1/q). \tag{C.10b}$$

By Conjecture III.1, $\widehat{v}_1$, for large enough $m_2\, n$,

$$\widehat{v}_1 = \beta_{\widehat{v},2\ell}\, \mathrm{vec}(bs^H) + \sqrt{\frac{1-\beta^2_{\widehat{v},2\ell}}{m_2\, n}}\, g$$

which, due to $\mathrm{reshape}_{m_2\, n\times 1\to m_2\times n}(\widehat{v}_1)$ in Step 4, gives us

$$\mathrm{reshape}(\widehat{v}^*) = \beta_{\widehat{v},2\ell}\, bs^H + \sqrt{\frac{1-\beta^2_{\widehat{v},2\ell}}{m_2\, n}}\, \mathrm{reshape}(g). \tag{C.11}$$

Equation (C.11) has rank one signal plus noise structure; rescaling it so that it has identity covariance, we get

$$\frac{\sqrt{m_2}\,\mathrm{reshape}(\widehat{v}_1)}{\sqrt{1-\beta^2_{\widehat{v},2\ell}}} = \underbrace{\frac{\sqrt{\beta^2_{\widehat{v},2\ell} m_2}}{\sqrt{1-\beta^2_{\widehat{v},2\ell}}}}_{:=\sigma_{b,2\ell}} bs^H + \frac{1}{\sqrt{n}}\mathrm{reshape}(g). \tag{C.12}$$

Substituting for $\beta_{\widehat{v},2\ell}$ from (C.10b), we have that

$$\sigma_{b,2\ell}^2 = \frac{\beta_{\widehat{v},2\ell}^2 m_2}{1-\beta_{\widehat{v},2\ell}^2} = \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2/m_2+1)} \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}. \tag{C.13}$$

By applying Proposition III.2 to (C.12), and using (C.13) to simplify, we get

$$|\langle \widehat{b}_{\mathsf{kpca2},2\ell}, b\rangle|^2 \xrightarrow{\text{a.s.}} \alpha^2\left(\sqrt{\frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}}, c_q\right). \tag{C.14}$$

Substituting (C.10a) and (C.14) in (3.7), we obtain (3.47b), which is the required result.

By Proposition III.2, through (C.10a), $\widehat{a}_{\mathsf{kpca2},1}$ is informative above the critical point given by

$$\sigma_0/\sqrt{m_2} = \left(\frac{m_1}{m_2\, n}\right)^{1/4}.$$

Therefore,

$$\sigma_{\mathsf{crit,akpca2},2\ell} = (m_1\, m_2/n)^{1/4} \to c_{\mathsf{svd}}^{1/4} \tag{C.15}$$

which is the same critical point as that of $\widehat{u}_{\mathsf{svd}}$. Similarly, from (C.12), the critical SNR for $\widehat{b}_{\mathsf{kpca2},2\ell}$ satisfies

$$\widehat{\sigma}_{\mathsf{b},2\ell} = \left(\frac{m_2}{n}\right)^{1/4} \implies \frac{\beta_{\widehat{v},2\ell}^2}{1-\beta_{\widehat{v},2\ell}^2} = \frac{1}{\sqrt{m_2\, n}}.$$

Substituting for $\beta_{\widehat{v},2\ell}$ from (C.14),

$$\frac{\sigma_0^4 - \dfrac{m}{n}}{m_2\left(\dfrac{m_2}{n}+\sigma_0^2\right)} = \frac{1}{\sqrt{m_2\, n}} \implies \sigma_0^4 - \sqrt{\frac{m_2}{n}}\sigma_0^2 - \left(\frac{m}{n} + \frac{m_1\sqrt{m_2}}{n\sqrt{n}}\right) = 0.$$

In the large array limit, this equation becomes

$$\sigma_0^4 - \sqrt{c_q}\sigma_0^2 - \left(c_{\mathsf{svd}} + c_p\sqrt{c_q}\right) = 0.$$

The above fourth degree equation has the following solution after discarding the negative and complex roots:

$$\sigma_{\mathsf{crit,bkpca2},2\ell} = \sqrt{\frac{\sqrt{c_q} + \sqrt{c_q + 4c_p\sqrt{c_q} + 4c_{\mathsf{svd}}}}{2}}, \tag{C.16}$$

which is the critical SNR for $\widehat{b}_{\mathsf{kpca2},2\ell}$. Since $|\widehat{u}_{\mathsf{kpca2},2\ell}^H u|^2 = 0$ if either $|\widehat{a}_{\mathsf{kpca2},2\ell}^H a|^2 = 0$ or

$|\widehat{b}^H_{\mathsf{kpca2},2\ell} b|^2 = 0$, we have that

$$\begin{aligned}\sigma_{\mathsf{crit,kpca2},2\ell} &= \max\{\sigma_{\mathsf{crit,akpca2},2\ell}, \sigma_{\mathsf{crit,bkpca2},2\ell}\} \\ &= \max\left\{c_{\mathsf{svd}}^{1/4}, \sqrt{\frac{\sqrt{c_q} + \sqrt{c_q + 4c_p\sqrt{c_q} + 4c_{\mathsf{svd}}}}{2}}\right\}.\end{aligned}$$

C.3 Analysis of KPCA2,2r

The analysis of KPCA2,2r is very similar to that of KPCA2,2ℓ. The rearrangement from Step 1, $\mathcal{R}_{m_1\, m_2\times n\to m_1\, n\times m_2}(Y)$ gives us

$$Y_{\mathcal{R}} = \sigma_0 \operatorname{vec}(as^H)\, b^T + \mathcal{R}(Z) \tag{C.17}$$

which has the rank one signal plus noise structure. We rescale it so that the noise matrix has unit variance:

$$\sqrt{\frac{n}{m_2}}\, Y_{\mathcal{R}} = \sqrt{\frac{n}{m_2}}\, \sigma_0 \operatorname{vec}\left(as^H\right)\, b^T + \sqrt{\frac{n}{m_2}}\mathcal{R}(Z).$$

By application of Proposition III.2, we have that

$$|\langle \widehat{u}_1, u\rangle|^2 = \alpha^2_{\widehat{u},2r} = \frac{\sigma_0^4 - m/n}{\sigma_0^2(\sigma_0^2 + m_1)} = O(1/m_1), \tag{C.18a}$$

and

$$|\langle \widehat{b}_{\mathsf{kpca2,2r}}, b\rangle|^2 = \frac{\sigma_0^4 - m/n}{\sigma_0^2(\sigma_0^2 + m_2/n)} \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_q)}. \tag{C.18b}$$

By Conjecture III.1, $\widehat{u}_1$ can be approximated as true vector plus Gaussian fluctuations for large enough $m_1\, n$, thus:

$$\widehat{u} = \alpha_{\widehat{u},2r} \operatorname{vec}(as^H) + \sqrt{\frac{1 - \alpha^2_{\widehat{u},2r}}{m_1\, n}}\, g.$$

Reshaping and rescaling this equation, similar to (C.11) through (C.12), we get

$$\sqrt{\frac{m_1}{1 - \alpha^2_{\widehat{u},2r}}} \operatorname{reshape}(\widehat{u}) = \underbrace{\alpha_{\widehat{u},2r} \sqrt{\frac{m_1}{1 - \alpha^2_{\widehat{u},2r}}}}_{:=\sigma_{a,2r}}\, a\, s^H + \sqrt{\frac{1}{n}} \operatorname{reshape}(g). \tag{C.19}$$

Substituting for $\alpha_{\widehat{u},2r}$ from (C.18a), we have that

$$\sigma_{a,2r}^2 = \frac{\alpha_{\widehat{u},2r}^2 m_2}{1-\alpha_{\widehat{u},2r}^2} = \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2 + c_{\mathsf{svd}}/m_1} \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}. \tag{C.20}$$

By Proposition III.2, and using (C.20) to simplify, we have that

$$|\langle \widehat{a}_{\mathsf{kpca2,2r}}, a\rangle|^2 \xrightarrow{\text{a.s.}} \alpha^2 \left(\frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2}, c_p \right). \tag{C.21}$$

Using (C.21) and (C.18b) in (3.7), we obtain (3.47c), which is the required result.

We now compute the critical SNR for algorithm estimates. From Proposition III.2 and (C.18b), the critical SNR for $\widehat{b}_{\mathsf{kpca2,2r}}$ satisfies

$$\sigma_0 \sqrt{\frac{n}{m_2}} = \left(\frac{m_1\, n}{m_2} \right)^{1/4}$$

which implies that

$$\sigma_{\mathsf{crit,bkpca2,2r}} = c_{\mathsf{svd}}^{1/4} \tag{C.22}$$

which the same as for $\widehat{u}_{\mathsf{svd}}$. Similarly, from (C.19), the critical SNR for $\widehat{a}_{\mathsf{kpca2,2r}}$ satisfies the equation

$$\widehat{\sigma}_{\mathsf{u},2r} = (m_1/n)^{1/4}.$$

Using the same approach as for KPCA,2ℓ, in the large array limit,

$$\sigma_0^4 - \sqrt{c_p}\sigma_0^2 - \left(c_{\mathsf{svd}} + c_q \sqrt{c_p} \right) = 0. \tag{C.23}$$

The above fourth degree equation has the following solution after discarding the negative and complex roots:

$$\sigma_{\mathsf{crit,akpca2,2r}} = \sqrt{\frac{\sqrt{c_p} + \sqrt{c_p + 4c_q\sqrt{c_p} + 4c_{\mathsf{svd}}}}{2}}, \tag{C.24}$$

which is the critical SNR for $\widehat{a}_{\mathsf{kpca2,2r}}$. The behavior of $\sigma_{\mathsf{crit,kpca2r}}$ is similar to that of $\sigma_{\mathsf{crit,kpca2}\ell}$, and the discussion at the end of Section III.4.2 may be repeated here, with m_1 and m_2 interchanged. As in KPCA2,2ℓ, we have that

$$\sigma_{\mathsf{crit,kpca2,2r}} = \max\{\sigma_{\mathsf{crit,akpca2,2r}}, \sigma_{\mathsf{crit,bkpca2,2r}}\}$$

$$= \max\left\{ c_{\mathsf{svd}}^{1/4}, \sqrt{\frac{\sqrt{c_p} + \sqrt{c_p + 4c_q\sqrt{c_p} + 4c_{\mathsf{svd}}}}{2}} \right\}.$$

C.4 Analysis of KPCA2,3

Recall that $\widehat{a}_{\mathsf{kpca2,3}} = \widehat{a}_{\mathsf{kpca2,2}\ell}$, and $\widehat{b}_{\mathsf{kpca2,3}} = \widehat{b}_{\mathsf{kpca2,2r}}$. Therefore, the derivations from (C.8) through (C.10), and (C.17) through (C.18) hold. Therefore,

$$|\langle \widehat{a}_{\mathsf{kpca2,3}}, a\rangle|^2 = \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_p)}, \tag{C.25}$$

$$|\langle \widehat{b}_{\mathsf{kpca2,3}}, b\rangle|^2 = \frac{\sigma_0^4 - c_{\mathsf{svd}}}{\sigma_0^2(\sigma_0^2 + c_q)}. \tag{C.26}$$

Using (C.25) and (C.26) in (3.7), we obtain (3.47d), which is the required result. The critical SNR of $\widehat{a}_{\mathsf{kpca2,3}}$ is given by (C.15) and that of $\widehat{b}_{\mathsf{kpca2,3}}$ is given by (C.22), which are both equal to $c_{\mathsf{svd}}^{1/4}$. Therefore, the critical SNR for the estimate $\widehat{u}_{\mathsf{kpca2,3}}$ is $c_{\mathsf{svd}}^{1/4}$.

APPENDIX D

Proof of Claim IV.1

We present the proof for KPCA3, 1, KPCA3, 3 and KPCA3, 6. We omit the proof of KPCA3, 2, since it is extremely similar to that of KPCA3, 1; of KPCA4, KPCA5 and KPCA8 since they are extremely similar to KPCA3, 6; and of KPCA3, 7 because it is extremely similar to KPCA3, 3.

D.1 Analysis of KPCA3, 1

In Step 1, we compute $\widehat{u}_{\text{svd}}$, the principal left singular vector of X. The performance of $\widehat{u}_{\text{svd}}$ is completely characterized by (4.11a). In Steps 2-3, we compute the SVD of $\mathcal{R}(\widehat{u}_{\text{svd}})$. From (4.15a),

$$\widehat{u}_{\text{svd}} = \alpha_{\text{svd}}(a \otimes b \otimes c) + \frac{\sqrt{1-\alpha_{\text{svd}}^2}}{\sqrt{m}} g.$$

Upon applying the rearrangement operator, we obtain

$$\mathcal{R}(\widehat{u}_{\text{svd}}) = \alpha_{\text{svd}} a(b \otimes c)^T + \frac{\sqrt{1-\alpha_{\text{svd}}^2}}{\sqrt{m}} \mathcal{R}(g). \tag{D.1}$$

Equation (D.1) describes a rank-one-signal plus noise model, similar to (4.1). However, the covariance of the noise component, $\mathcal{R}(g)$ of (D.1), does not scale with the sample size as in (4.1). In order to apply the results of Proposition IV.1 to (D.1), we rescale the equation

appropriately:

$$\frac{\sqrt{m_1}}{\sqrt{1-\alpha_{\mathsf{svd}}^2}}\mathcal{R}(\widehat{u}_{\mathsf{svd}}) = \underbrace{\frac{\sqrt{m_1\alpha_{\mathsf{svd}}^2}}{\sqrt{1-\alpha_{\mathsf{svd}}^2}}}_{:=\sigma_{1,1}} a(b\otimes c)^T + \frac{1}{\sqrt{m_2m_3}}\mathcal{R}(g). \tag{D.2}$$

Therefore, we have a rank-one signal plus noise model with eigen-SNR $\sigma_{1,1}$ and "aspect ratio" $c_{1,1} := m_1/(m_2m_3)$. From Proposition IV.1, we have that

$$|\langle \widehat{a}_{\mathsf{kpca3},1}, a\rangle|^2 = \alpha^2(\sigma_{1,1}, c_{1,1}), \tag{D.3a}$$

$$|\langle \widehat{v}, b\otimes c\rangle|^2 = \beta^2(\sigma_{1,1}, c_{1,1}) := \beta_{1,1}^2. \tag{D.3b}$$

Plugging in the definitions of $\sigma_{1,1}$ and $c_{1,1}$ from (D.2), and computing (D.3a) explicitly using (4.10a), we get

$$|\langle \widehat{a}_{\mathsf{kpca3},1}, a\rangle|^2 = \frac{\left(\frac{\alpha_{\mathsf{svd}}^2}{1-\alpha_{\mathsf{svd}}^2}\right)^2 - \frac{1}{m}}{\left(\frac{\alpha_{\mathsf{svd}}^2}{1-\alpha_{\mathsf{svd}}^2}\right)\left(\frac{\alpha_{\mathsf{svd}}^2}{1-\alpha_{\mathsf{svd}}^2} + \frac{1}{m_2m_3}\right)},$$

and from (4.14a), and (4.10a),

$$= \frac{(\sigma_0^4 - c_0)^2 - \frac{c_0^2(1+\sigma_0^2)^2}{m}}{(\sigma_0^4 - c_0)\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2m_3}\right)}.$$

In the large array limit, as $m_1, m_2, m_3 \to \infty$, we have that

$$|\langle \widehat{a}_{\mathsf{kpca3},1}, a\rangle|^2 \to 1. \tag{D.4}$$

Similarly, consider Steps 4 - 7. Using the approximation (4.15b) and (D.3b), we have that

$$\widehat{v} = \beta_{1,1}(b\otimes c) + \frac{\sqrt{1-\beta_{1,1}^2}}{\sqrt{m_2m_3}}g,$$

$$\frac{\sqrt{m_2}}{\sqrt{1-\beta_{1,1}^2}}\mathcal{R}(\widehat{v}) = \underbrace{\frac{\sqrt{m_2\beta_{1,1}^2}}{\sqrt{1-\beta_{1,1}^2}}}_{:=\sigma_{1,2}} bc^T + \frac{1}{\sqrt{m_3}}\mathcal{R}(g), \tag{D.5}$$

where we have appropriately rescaled the rank-one-signal plus noise model resulting from the rearrangement $\widehat{v} \to \mathcal{R}(\widehat{v})$. Therefore, we can apply the results of (4.11) to (D.5), with eigen-SNR $\sigma = \sigma_{1,1}$ and aspect ratio $c_{1,2} := m_2/m_3$. We have that,

$$|\langle \widehat{b}_{\mathsf{kpca3,1}}, b\rangle|^2 = \alpha^2(\sigma_{1,2}, c_{1,2}), \tag{D.6a}$$

$$|\langle \widehat{c}_{\mathsf{kpca3,1}}, c\rangle|^2 = \beta^2(\sigma_{1,2}, c_{1,2}). \tag{D.6b}$$

Substituting for $\sigma_{1,2}$ and $c_{1,2}$ from (D.5) into (4.10), we have that

$$|\langle \widehat{b}_{\mathsf{kpca3,1}}, b\rangle|^2 = \frac{m_2^2\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}\right)^2 - \frac{m_2}{m_3}}{\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}\right)\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}+\frac{1}{m_3}\right)}, \tag{D.7a}$$

$$|\langle \widehat{c}_{\mathsf{kpca3,1}}, c\rangle|^2 = \frac{m_2^2\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}\right)^2 - \frac{m_2}{m_3}}{\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}\right)\left(\frac{\beta_{1,1}^2}{1-\beta_{1,1}^2}+\frac{1}{m_2}\right)}. \tag{D.7b}$$

Using the definitions of $\sigma_{1,1}$ and $c_{1,1}$ from (D.2) in (4.10a), we have that

$$\sigma_{1,2} = \frac{\sqrt{(\sigma_0^4 - c_0)^2 - \frac{c_0^2(1+\sigma_0^2)}{m}}}{\sqrt{\frac{c_0(1+\sigma_0^2)}{m_1 m_2}\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2 m_3}\right)}}. \tag{D.8}$$

Using (D.8), we have that

$$|\langle \widehat{b}_{\mathsf{kpca3,1}}, b\rangle|^2 = \frac{\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m}\right]^2 - \frac{c_0(1+\sigma_0^2)}{mm_1}\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2 m_3}\right)^2}{\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m}\right]\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m} + \frac{c_0(1+\sigma_0^2)}{m_1 m_3}\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2 m_3}\right)\right]},$$

$$|\langle \widehat{c}_{\mathsf{kpca3,1}}, b\rangle|^2 = \frac{\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m}\right]^2 - \frac{c_0(1+\sigma_0^2)}{mm_1}\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2m_3}\right)^2}{\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m}\right]\left[(\sigma_0^4 - c_0)^4 - \frac{c_0^2(1+\sigma_0^2)^2}{m} + \frac{c_0(1+\sigma_0^2)}{m_1m_2}\left(\sigma_0^4 - c_0 + \frac{c_0(1+\sigma_0^2)}{m_2m_3}\right)\right]}.$$

In the large array limit, as $m_1, m_2, m_3 \to \infty$,

$$|\langle \widehat{b}_{\mathsf{kpca3,1}}, b\rangle|^2 \to 1, \tag{D.9a}$$

$$|\langle \widehat{c}_{\mathsf{kpca3,1}}, c\rangle|^2 \to 1. \tag{D.9b}$$

Now we compute the critical SNR, analogous to (4.12), for the estimates $\widehat{a}_{\mathsf{kpca3,1}}$, $\widehat{b}_{\mathsf{kpca3,1}}$ and $\widehat{c}_{\mathsf{kpca3,1}}$. Using Proposition IV.1 in (D.2), see that the critical SNR for $\widehat{a}_{\mathsf{kpca3,1}}$ is the value of SNR which solves

$$\sigma_{1,1} = c_{1,1}^{1/4}.$$

Substituting for $\sigma_{1,1}$ from (D.2), we obtain

$$\frac{\alpha_{\mathsf{svd}}^2}{1 - \alpha_{\mathsf{svd}}^2} = \frac{1}{\sqrt{m}}, \tag{D.10}$$

and substituting for α_{svd} in terms of σ_0 and c_0 from (4.14a), we get the equation

$$\sigma^4 - \frac{c}{\sqrt{m}}\sigma^2 - c_0\left(1 + \frac{1}{\sqrt{m}}\right) = 0. \tag{D.11}$$

In the large array limit as $m \to \infty$, (D.11) reduces to

$$\sigma^4 - c_0 = 0,$$

which admits only one valid solution ($\sigma_0 \in \mathbb{R}_+$), so that the critical SNR for $\widehat{a}_{\mathsf{kpca3,1}}$ in the large array limit is

$$\sigma_{\mathsf{crit,a,1}} \to c_0^{1/4}. \tag{D.12}$$

Similarly, using Proposition IV.1 in (D.5), the critical SNR for $\widehat{b}_{\mathsf{kpca3,1}}$ and $\widehat{c}_{\mathsf{kpca3,1}}$ solve the equation

$$\sigma_{2,1} = c_{2,1}^{1/4}.$$

Substituting for $\sigma_{2,1}$ and $c_{2,1}$ from (D.5), we have that

$$\frac{\beta_{1,1}^2}{1 - \beta_{1,1}^2} = \frac{1}{\sqrt{m_2m_3}}.$$

Substituting for α_{svd} in the resulting expression from (4.14a), we obtain

$$(\sigma^4 - c_0)^2 - \frac{(\sigma^4 - c_0)(1+\sigma^2)c_0}{m\sqrt{m_2 m_3}} - \frac{c_0^2(1+\sigma^2)^2}{m}\left(1 + \frac{1}{(m_2 m_3)^{1/3}}\right) = 0,$$

which, in the large array limit, reduces to

$$(\sigma^4 - c_0)^2 = 0.$$

Therefore,

$$\sigma_{\mathsf{crit,b},1} = c_0^{1/4}, \tag{D.13a}$$
$$\sigma_{\mathsf{crit,c},1} = c_0^{1/4}. \tag{D.13b}$$

D.2 Analysis of KPCA$3,3$

Consider Steps $1-4$. Upon rescaling $\mathcal{R}(X)$ appropriately, we get

$$\frac{\mathcal{R}(X)}{\sqrt{m_3}} = \underbrace{\frac{\sigma_0}{\sqrt{m_3}}}_{:=\sigma_{3,1}}(a\otimes b)\mathrm{vec}(cs^H)^T + \frac{\mathcal{R}(Z)}{\sqrt{m_3}}. \tag{D.14}$$

This is a rank-one signal plus noise model, with eigen-SNR $\sigma_{3,1}$ and aspect ratio $c_{3,1} := (m_1 m_2)/(m_3 n)$. Using Proposition IV.1, we have that

$$|\langle \widehat{u}, a\otimes b\rangle|^2 = \alpha^2(\sigma_{3,1}, c_{3,1}) := \alpha_{3,1}^2, \tag{D.15a}$$
$$|\langle \widehat{v}, \mathrm{vec}(cs^H)\rangle|^2 = \beta^2(\sigma_{3,1}, c_{3,1}) := \beta_{3,1}^2. \tag{D.15b}$$

Consider Steps $5-6$. Using (4.15a), and rescaling the resulting equation appropriately, we have that

$$\frac{\sqrt{m_1}}{\sqrt{1-\alpha_{3,1}^2}}\mathcal{R}(\widehat{u}) = \underbrace{\frac{\sqrt{m_1\alpha_{3,1}^2}}{\sqrt{1-\alpha_{3,1}^2}}}_{:=\sigma_{3,2}} ab^T + \frac{1}{\sqrt{m_2}}\mathcal{R}(g). \tag{D.16}$$

This is a rank-one signal plus noise model with eigen-SNR $\sigma_{3,2}$ and aspect ratio $c_{3,2} = m_1/m_2$. Using Proposition IV.1, we have that

$$|\langle \widehat{a}_{\mathsf{kpca3,3}}, a\rangle|^2 = \alpha^2(\sigma_{3,2}, c_{3,2}), \tag{D.17a}$$
$$|\langle \widehat{b}_{\mathsf{kpca3,3}}, b\rangle|^2 = \beta^2(\sigma_{3,2}, c_{3,2}). \tag{D.17b}$$

Consider Steps 7 – 9. By appropriately rescaling the equation resulting from reshape$(\widehat{v})$, we have that

$$\frac{\sqrt{m_3}}{\sqrt{1-\beta_{3,1}^2}}\text{reshape}(\widehat{v}) = \underbrace{\frac{\sqrt{m_3\beta_{3,1}^2}}{\sqrt{1-\beta_{3,1}^2}}}_{:=\sigma_{3,3}} cs^H + \frac{1}{\sqrt{n}}\text{reshape}(g). \tag{D.18}$$

Using Proposition IV.1, we have that

$$|\langle \widehat{c}_{\mathsf{kpca3,3}}, c\rangle|^2 = \alpha^2(\sigma_{3,3}, c_{3,3}). \tag{D.19}$$

By using (4.10a) and the definition of $\sigma_{3,2}$ from (D.16), we have that

$$\sigma_{3,2} = \frac{\sqrt{\sigma_0^4 - \dfrac{m}{n}}}{\sqrt{\dfrac{m_2 m_3}{n}\left(1 + \dfrac{\sigma_0^2}{m_3}\right)}}. \tag{D.20}$$

Using this to compute (D.17) explicitly,

$$|\langle \widehat{a}_{\mathsf{kpca3,2}}, a\rangle|^2 = \frac{\left(\sigma_0^4 - \dfrac{m}{n}\right)^2 - \dfrac{m_1 m_2 m_3^2}{n^2}\left(1 + \dfrac{\sigma_0^2}{m_3}\right)^2}{\left(\sigma_0^4 - \dfrac{m}{n}\right)\left(\sigma_0^4 - \dfrac{m}{n} + \dfrac{m_1 m_3}{n}\left(1 + \dfrac{\sigma_0^2}{m_3}\right)\right)},$$
$$|\langle \widehat{b}_{\mathsf{kpca3,2}}, b\rangle|^2 = \frac{\left(\sigma_0^4 - \dfrac{m}{n}\right)^2 - \dfrac{m_1 m_2 m_3^2}{n^2}\left(1 + \dfrac{\sigma_0^2}{m_3}\right)^2}{\left(\sigma_0^4 - \dfrac{m}{n}\right)\left(\sigma_0^4 - \dfrac{m}{n} + \dfrac{m_2 m_3}{n}\left(1 + \dfrac{\sigma_0^2}{m_3}\right)\right)}.$$

In the large array limit, we have that

$$|\langle \widehat{a}_{\mathsf{kpca3,2}}, a\rangle|^2 \xrightarrow{\text{a.s.}} \frac{(\sigma_0 - c_0)^2 - c_0 c_3}{(\sigma_0 - c_0)^2}, \tag{D.22a}$$

$$|\langle \widehat{b}_{\mathsf{kpca3,2}}, b\rangle|^2 \xrightarrow{\text{a.s.}} \frac{(\sigma_0 - c_0)^2 - c_0 c_3^2}{(\sigma_0 - c_0)^2}. \tag{D.22b}$$

Similarly, using (4.10b) and the definition of $\sigma_{3,3}$ from (D.18), we have that

$$\sigma_{3,3} = \frac{\sqrt{\sigma_0^4 - \dfrac{m}{n}}}{\sqrt{\sigma_0^2 + \dfrac{m_1 m_2}{n}}}. \tag{D.23}$$

Using this in (D.19), we get

$$|\langle \widehat{c}_{\mathsf{kpca3,3}}, c\rangle|^2 = \frac{\left(\sigma_0^4 - \dfrac{m}{n}\right)^2 - \dfrac{m_3}{n}\left(\sigma_0^2 + \dfrac{m_1 m_2}{n}\right)^2}{\left(\sigma_0^4 - \dfrac{m}{n}\right)\left(\sigma_0^4 - \dfrac{m}{n} + \dfrac{m_3}{n}\left(\dfrac{m_1 m_2}{n} + \sigma_0^2\right)\right)}.$$

In the large array limit, we have that,

$$|\langle \widehat{c}_{\mathsf{kpca3,3}}, c\rangle|^2 = \frac{(\sigma_0^4 - c_0)^2 - c_3(\sigma_0^2 + c_{12})^2}{(\sigma_0^4 - c_0)(\sigma_0^4 - c_0 + c_3(\sigma_0^2 + c_{12}))}. \tag{D.24}$$

Now we compute the critical SNRs for $\widehat{a}_{\mathsf{kpca3,3}}$, $\widehat{b}_{\mathsf{kpca3,3}}$ and $\widehat{c}_{\mathsf{kpca3,3}}$. From (D.16) and (D.17), we see that the $\sigma_{\mathsf{crit,a,3}}$ and $\sigma_{\mathsf{crit,b,3}}$ satisfy

$$\sigma_{3,2} = c_{3,2}^{1/4}.$$

Using (D.20), we have

$$\sigma^4 - \frac{m}{n}\left(1 + \frac{1}{\sqrt{m_1 m_2}}\right) - \frac{\sigma^2 m}{\sqrt{m_1 m_2} m_3 n} = 0,$$

which, in the large array limit, converges to

$$\sigma^4 - c_0 = 0.$$

Therefore,

$$\sigma_{\mathsf{crit,a,3}} = c_0^{1/4}, \tag{D.25a}$$

$$\sigma_{\mathsf{crit,b,3}} = c_0^{1/4}. \tag{D.25b}$$

Similarly, from (D.18) we see that the critical SNR for $\widehat{c}_{\mathsf{kpca3,3}}$ satisfies

$$\sigma_{3,3} = c_{3,3}^{1/4}.$$

Using (D.23), we get

$$\sigma_0^4 - \frac{m}{n} - \frac{\sqrt{m_3}}{\sqrt{n}} - \frac{\sqrt{m_3}}{\sqrt{n}m_1m_2}\sigma^2 = 0,$$

which in the large array limit converges to

$$\sigma^4 - c - \sqrt{c_3} = 0.$$

Therefore,

$$\sigma_{\mathsf{crit,c,3}} = (c_0 + \sqrt{c_3})^{1/4}. \tag{D.26}$$

D.3 Analysis of KPCA$3,6$

Consider Steps $1-2$. Upon appropriate rescaling of $\mathcal{R}(X)$, we have that

$$\frac{\mathcal{R}(X)}{\sqrt{m_2m_3}} = \underbrace{\frac{\sigma_0}{\sqrt{m_2m_3}}}_{:=\sigma_{6,1}} a\,\mathrm{vec}((b\otimes c)s^H)^T + \frac{\mathcal{R}(Z)}{\sqrt{m_2m_3}}. \tag{D.27}$$

This is a rank-one signal plus noise model with eigen-SNR $\sigma_{6,1}$ and aspect ratio $c_{6,1} = (m_1)/(m_2m_3n)$. Therefore, by Proposition IV.1 we have that

$$|\langle \widehat{a}_{\mathsf{kpca3,6}}, a\rangle|^2 = \alpha^2(\sigma_{6,1}, c_{6,1}) := \alpha_{6,1}^2, \tag{D.28a}$$

$$|\langle \widehat{v}, \mathrm{vec}((b\otimes c)s^H)\rangle| = \beta^2(\sigma_{6,1}, c_{6,1}) := \beta_{4,1}^2. \tag{D.28b}$$

Consider Steps $3-6$. We have from (4.15b) that

$$(\widehat{v}) = \beta_{6,1}\mathrm{vec}((b\otimes c)s^H) + \frac{\sqrt{1-\beta_{6,1}^2}}{\sqrt{m_2m_3n}}g.$$

After application of the rearrangement operator and appropriate rescaling,

$$\frac{\sqrt{m_2 n}}{\sqrt{1-\beta_{6,1}^2}}\mathcal{R}(\widehat{v}) = \underbrace{\frac{\sqrt{m_2 n \beta_{6,1}^2}}{\sqrt{1-\beta_{6,1}^2}}}_{:=\sigma_{6,2}} \mathrm{vec}(bs^H)c^T + \frac{1}{\sqrt{m_3}}\mathcal{R}(g). \tag{D.29}$$

This is a rank-one signal plus noise model with eigen-SNR $\sigma_{6,2}$ and aspect ratio $c_{6,2} = (m_2 n/m_3)$. From Proposition IV.1,

$$|\langle \widehat{u}, \mathrm{vec}(bs^H)\rangle|^2 = \alpha^2(\sigma_{6,2}, c_{6,2}) := \alpha_{6,2}^2, \tag{D.30a}$$

$$|\langle \widehat{c}_{\mathsf{kpca3,6}}, c\rangle|^2 = \beta^2(\sigma_{6,2}, c_{6,2}) := \beta_{6,2}^2. \tag{D.30b}$$

Consider Steps $7-9$. We have from (4.15a) that

$$\widehat{u} = \alpha_{6,2}\mathrm{vec}(bs^H) + \frac{\sqrt{1-\alpha_{6,2}^2}}{\sqrt{m_2 n}} g.$$

Upon application of the reshape operator and appropriate rescaling, we have that

$$\frac{\sqrt{m_2}}{\sqrt{1-\alpha_{6,2}^2}}\mathrm{reshape}(\widehat{u}) = \underbrace{\frac{\sqrt{m_2 \alpha_{6,2}^2}}{\sqrt{1-\alpha_{6,2}^2}}}_{:=\sigma_{6,3}} bs^H + \frac{1}{\sqrt{n}}\mathrm{reshape}(g). \tag{D.31}$$

This is a rank-one signal plus noise model with eigen-SNR $\sigma_{6,3}$ and aspect ratio $c_{6,3} = m_2/n$. We have from Proposition IV.1 that

$$|\langle \widehat{b}_{\mathsf{kpca3,6}}, b\rangle|^2 = \alpha^2(\sigma_{6,3}, c_{6,3}). \tag{D.32}$$

We now compute (D.28a), (D.32) and (D.30b) explicitly. From (D.27) and (4.10a), (D.28a) becomes

$$|\langle \widehat{a}_{\mathsf{kpca3,6}}, a\rangle|^2 = \frac{\sigma_0^4 - \dfrac{m}{n}}{\sigma_0^2\left(\sigma_0^2 + \dfrac{m_1}{n}\right)}.$$

In the large array limit, we have that

$$|\langle \widehat{a}_{\mathsf{kpca3,6}}, a\rangle|^2 \xrightarrow{\text{a.s.}} \frac{\sigma_0^4 - c_0}{\sigma_0^2(\sigma_0^2 + c_1)}. \tag{D.33}$$

Using (4.10b) in (D.29), we have that

$$\sigma_{6,2} = \frac{\sqrt{\sigma_0^4 - \frac{m}{n}}}{\sqrt{\frac{m_3}{n}\left(\sigma_0^2 + \frac{m_1}{n}\right)}}. \tag{D.34}$$

Using (D.34) and (4.10b) in (D.30b), we have that

$$|\langle \widehat{c}_{\mathsf{kpca3,6}}, c\rangle|^2 = \frac{\left(\sigma_0^4 - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2}{\left(\sigma_0^4 - \frac{m}{n}\right)\left(\sigma_0^4 - \frac{m}{n} + \frac{m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)\right)},$$

so that in the large array limit, we have that

$$|\langle \widehat{c}_{\mathsf{kpca3,6}}, c\rangle|^2 \xrightarrow{\text{a.s.}} \frac{(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2}{(\sigma_0^4 - c)\left(\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right)}. \tag{D.35}$$

Similarly, expanding out $\sigma_{6,3}$ from (D.31) using (4.10a), and using (D.34) gives us

$$\sigma_{6,3} = \frac{\sqrt{\left(\sigma_0^4 - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2}}{\sqrt{\left(\frac{m_1}{n} + \sigma_0^2\right)\left[\frac{m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right) + \sigma_0^4 - \frac{m}{n}\right]}}. \tag{D.36}$$

Using (D.36) and (4.10a), we get (D.37).

$$|\langle \widehat{b}_{\mathsf{kpca3,6}}, b\rangle|^2 = \frac{\widetilde{T}_{6b,1} - \widetilde{T}_{6b,2}}{\widetilde{T}_{6b,3} \cdot \widetilde{T}_{6b,4}}. \tag{D.37}$$

$$\widetilde{T}_{6b,1} = \left[\left(\sigma_0^4 - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2\right]^2, \tag{D.38}$$

$$\widetilde{T}_{6b,2} = \frac{m_2}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2\left[\sigma_0^4 - \frac{m}{n} + \frac{m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)\right]^2, \tag{D.39}$$

$$\widetilde{T}_{6b,3} = \left[\left(\sigma_0^4 - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2\right] \tag{D.40}$$

$$\widetilde{T}_{6b,4} = \left[\left(\sigma_0^4 - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)^2 + \frac{m_2}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)\left[\sigma_0^4 - \frac{m}{n} + \frac{m_2}{n}\left(\frac{m_1}{n} + \sigma_0^2\right)\right]\right]. \tag{D.41}$$

In the large array limit, we have from (D.37) that

$$|\langle \widehat{b}_{\mathsf{kpca3,6}}, b\rangle|^2 \xrightarrow{\text{a.s.}} \frac{T_{6b,1} - T_{6b,2}}{T_{6b,3}T_{6b,4}}. \tag{D.42}$$

where

$$\begin{aligned}
T_{6b,1} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right]^2, \\
T_{6b,2} &= c_2(c_1 + \sigma_0^2)^2\left[\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right]^2, \\
T_{6b,3} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2\right], \\
T_{6b,4} &= \left[(\sigma_0^4 - c_0)^2 - c_{23}(c_1 + \sigma_0^2)^2 + c_2(c_1 + \sigma_0^2)\left[\sigma_0^4 - c_0 + c_3(c_1 + \sigma_0^2)\right]\right].
\end{aligned}$$

The critical SNR for $\widehat{a}_{\mathsf{kpca3,6}}$ solves

$$\begin{aligned}
\sigma_{6,1} &= c_{6,1}^{1/4}, \\
\frac{\sigma}{\sqrt{m_2 m_3}} &= \left(\frac{m_1}{m_2 m_3 n}\right)^{1/4},
\end{aligned}$$

so that in the large array limit,

$$\sigma_{\mathsf{crit,a,6}} = c_0^{1/4}. \tag{D.43}$$

The critical SNR for $\widehat{c}_{\mathsf{kpca3,6}}$ solves

$$\sigma_{6,2} = c_{6,2}^{1/4}$$

and from (D.34),

$$\sigma^4 - \frac{m}{n} = \frac{\sqrt{m_2 m_3}}{\sqrt{n}}\left(\frac{m_1}{n} + \sigma^2\right).$$

Therefore, in the large array limit, we have that

$$\sigma_{\mathsf{crit,c,6}} = \left(\frac{\sqrt{c_{23}} + \sqrt{c_2 3 + 4c + 4\sqrt{c_{23}}c_1}}{2}\right)^{1/2}. \tag{D.44}$$

The critical SNR for $\widehat{b}_{\mathsf{kpca3,6}}$ solves

$$\sigma_{6,3} = c_{6,3}^{1/4},$$

which from (D.36) implies,

$$\left(\sigma - \frac{m}{n}\right)^2 - \frac{m_2 m_3}{n}\left(\frac{m_1}{n} + \sigma^2\right)^2 = \frac{\sqrt{m_2}}{\sqrt{n}}\left(\frac{m_1}{n} + \sigma^2\right)\left[\frac{m_3}{n}\left(\frac{m_1}{n} + \sigma^2\right) + \sigma^4 - \frac{m}{n}\right].$$

In the large array limit, this equation reduces to

$$(\sigma^2 - c)^4 - c_{23}(c_1 + \sigma^2)^2 = \sqrt{c_2}(c_1 + \sigma^2)\left(c_3(c_1 + \sigma^2) + \sigma^4 - c\right). \tag{D.45}$$

Then, $\sigma_{\mathsf{crit,b,6}} = \kappa_{6,b}$, where $\kappa_{6,b}$ is the positive, real valued root of (D.45).

BIBLIOGRAPHY

[1] F. Li, H. Liu, and R. J. Vaccaro, "Performance analysis for doa estimation algorithms: unification, simplification, and observations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 4, pp. 1170–1184, Oct 1993.

[2] K. Greenewald, E. Zelnio, and A. O. Hero III, "Kronecker PCA Based Robust SAR STAP."

[3] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, Mar 1986.

[4] S. V. Schell, "Performance analysis of the cyclic music method of direction estimation for cyclostationary signals," *IEEE Transactions on Signal Processing*, vol. 42, no. 11, pp. 3043–3050, Nov 1994.

[5] M. L. McCloud and L. L. Scharf, "A new subspace identification algorithm for high-resolution doa estimation," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 10, pp. 1382–1390, Oct 2002.

[6] B. D. Rao and K. V. S. Hari, "Performance analysis of root-music," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 1939–1949, Dec 1989.

[7] M. D. Zoltowski, G. M. Kautz, and S. D. Silverstein, "Beamspace root-music," *IEEE Transactions on Signal Processing*, vol. 41, no. 1, pp. 344–, Jan 1993.

[8] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, Jul 1989.

[9] N. Yuen and B. Friedlander, "Asymptotic performance analysis of esprit, higher order esprit, and virtual esprit algorithms," *Signal Processing, IEEE Transactions on*, vol. 44, no. 10, pp. 2537–2550, Oct 1996.

[10] L. Balzano, B. Recht, and R. Nowak, "High-dimensional matched subspace detection when data are missing," in *2010 IEEE International Symposium on Information Theory*, June 2010, pp. 1638–1642.

[11] A. Ribeiro and G. B. Giannakis, "Bandwidth-constrained distributed estimation for wireless sensor networks-part i: Gaussian case," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1131–1143, March 2006.

[12] ——, "Bandwidth-constrained distributed estimation for wireless sensor networks-part ii: unknown probability density function," *IEEE Transactions on Signal Processing*, vol. 54, no. 7, pp. 2784–2796, July 2006.

[13] F. Benaych-Georges and R. R. Nadakuditi, "The eigenvalues and eigenvectors of finite, low rank perturbations of large random matrices," *Advances in Mathematics*, vol. 227, no. 1, pp. 494 – 521, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0001870811000570

[14] ——, "The singular values and vectors of low rank perturbations of large rectangular random matrices," *Journal of Multivariate Analysis*, vol. 111, pp. 120 – 135, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0047259X12001108

[15] R. Nadakuditi, "Optshrink: An algorithm for improved low-rank signal matrix denoising by optimal, data-driven singular value shrinkage," *Information Theory, IEEE Transactions on*, vol. 60, no. 5, pp. 3002–3018, May 2014.

[16] J. R. Guerci, *Space-time Adaptive Processing for Radar*. Artech House, 2003.

[17] J. Li, G. Liao, and H. Griffiths, "Bistatic mimo radar space-time adaptive processing," in *Radar Conference (RADAR), 2011 IEEE*, May 2011, pp. 498–502.

[18] A. J. Laub, *Matrix analysis for scientists and engineers*. Siam, 2005.

[19] C. F. Van Loan and N. Pitsianis, *Approximation with Kronecker Products*, ser. NATO ASI Series. Springer Netherlands, 1993, vol. 232.

[20] B. D. V. Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4–24, April 1988.

[21] D. J. Love, R. W. Heath, and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2735–2747, Oct 2003.

[22] T. Yoo and A. Goldsmith, "On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 528–541, March 2006.

[23] L. C. Godara, "Application of antenna arrays to mobile communications. ii. beamforming and direction-of-arrival considerations," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1195–1245, Aug 1997.

[24] H. Van Trees, *Optimum Array Processing*. Wiley Online Library, 2002, vol. 7.

[25] F. Li and R. Vaccaro, "Sensitivity analysis of doa estimation algorithms to sensor errors," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 28, no. 3, pp. 708–717, Jul 1992.

[26] A. Swindlehurst and T. Kailath, "A performance analysis of subspace-based methods in the presence of model errors. i. the music algorithm," *Signal Processing, IEEE Transactions on*, vol. 40, no. 7, pp. 1758–1774, 1992.

[27] P. Stoica and A. Nehorai, "Music, maximum likelihood, and cramer-rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[28] ——, "Music, maximum likelihood, and cramer-rao bound: further results and comparisons," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 12, pp. 2140–2150, Dec 1990.

[29] J.-F. Cardoso and E. Moulines, "Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants," *Signal Processing, IEEE Transactions on*, vol. 43, no. 1, pp. 214–224, Jan 1995.

[30] A. Swindlehurst and T. Kailath, "A performance analysis of subspace-based methods in the presence of model error. ii. multidimensional algorithms," *Signal Processing, IEEE Transactions on*, vol. 41, no. 9, pp. 2882–2890, Sep 1993.

[31] A. Ferreol, P. Larzabal, and M. Viberg, "On the asymptotic performance analysis of subspace doa estimation in the presence of modeling errors: case of music," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 907–920, March 2006.

[32] B. Friedlander, "A sensitivity analysis of the music algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1740–1751, Oct 1990.

[33] X. Mestre and F. Rubio, "An improved subspace-based algorithm in the small sample size regime," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 4, May 2006, pp. IV–IV.

[34] X. Mestre and M. Ã. Lagunas, "Modified subspace algorithms for doa estimation with large arrays," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 598–614, Feb 2008.

[35] P. Vallet, W. Hachem, P. Loubaton, X. Mestre, and J. Najim, "An improved music algorithm based on low rank perturbation of large random matrices," in *2011 IEEE Statistical Signal Processing Workshop (SSP)*, June 2011, pp. 689–692.

[36] D. Paul, "Asymptotics of sample eigenstructure for a large dimensional spiked covariance model," *Statistica Sinica*, vol. 17, no. 4, p. 1617, 2007.

[37] N. E. Karoui, "Spectrum estimation for large dimensional covariance matrices using random matrix theory," *The Annals of Statistics*, pp. 2757–2790, 2008.

[38] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5947–5959, Dec 2013.

[39] B. A. Johnson, Y. I. Abramovich, and X. Mestre, "Music, g-music, and maximum-likelihood performance breakdown," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3944–3958, Aug 2008.

[40] A. Onatski, "Asymptotics of the principal components estimator of large factor models with weakly influential factors," *Journal of Econometrics*, vol. 168, no. 2, pp. 244 – 258, 2012.

[41] A. Bloemendal, L. Erdos, A. Knowles, H.-T. Yau, and J. Yin, "Isotropic local laws for sample covariance and generalized wigner matrices," *Electron. J. Probab*, vol. 19, no. 33, pp. 1–53, 2014.

[42] A. Kangas, P. Stoica, and T. Söderström, "Large-sample analysis of music and min-norm direction estimators in the presence of model errors," *Circuits, Systems and Signal Processing*, vol. 15, no. 3, pp. 377–393, May 1996. [Online]. Available: http://dx.doi.org/10.1007/BF01182593

[43] J. Galambos, *The asymptotic theory of extreme order statistics*. Krieger Malabar, 1987.

[44] S. Boucheron, G. Lugosi, and P. Massart, *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[45] S. Negahban and M. J. Wainwright, "Restricted strong convexity and weighted matrix completion: Optimal bounds with noise," *J. Mach. Learn. Res.*, vol. 13, pp. 1665–1697, May 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2188385.2343697

[46] E. Candes and M. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[47] E. Candes and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *Information Theory, IEEE Transactions on*, vol. 56, no. 5, pp. 2053–2080, May 2010.

[48] E. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, June 2010.

[49] T.-J. Shan and T. Kailath, "Adaptive beamforming for coherent signals and interference," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 3, pp. 527–536, 1985.

[50] P. Vallet, "Random matrix theory and applications to statistical signal processing," Ph.D. dissertation, Ph. D. dissertation, Université de Paris-Est, Marne-la-Vallée, 2011.

[51] W. Hachem, P. Loubaton, X. Mestre, J. Najim, and P. Vallet, "Large information plus noise random matrix models and consistent subspace estimation in large sensor networks," *Random Matrices: Theory and Applications*, vol. 1, no. 02, 2012.

[52] E. J. Hannan, "The estimation of frequency," *Journal of Applied Probability*, vol. 10, no. 3, pp. pp. 510–519, 1973. [Online]. Available: http://www.jstor.org/stable/3212772

[53] T. Amemiya, *Advanced econometrics*. Harvard university press, 1985.

[54] K. Sharman, T. Durrani, M. Wax, and T. Kailath, "Asymptotic performance of eigenstructure spectral analysis methods," in *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, Mar 1984, pp. 440–443.

[55] W. L. Melvin, "A STAP overview," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, no. 1, pp. 19–35, 2004.

[56] T. Tsiligkaridis and A. O. Hero, "Covariance estimation in high dimensions via kronecker product expansions," *Signal Processing, IEEE Transactions on*, vol. 61, no. 21, pp. 5347–5360, Nov 2013.

[57] K. Greenewald and A. Hero, "Regularized block toeplitz covariance matrix estimation via kronecker product expansions," in *Statistical Signal Processing (SSP), 2014 IEEE Workshop on*, June 2014, pp. 9–12.

[58] J. Ritcey and A. Chindapol, "A Kronecker product improvement to PCA for space time adaptive processing," in *Signals, Systems and Computers, 2000. Conference Record of the Thirty-Fourth Asilomar Conference on*, vol. 1, Oct 2000, pp. 651–655 vol.1.

[59] A. P. da Silva, P. Comon, and A. L. F. de Almeida, "An iterative deflation algorithm for exact cp tensor decomposition," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 3961–3965.

[60] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "On the best rank-1 and rank-(r1 ,r2 ,. . .,rn) approximation of higher-order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000. [Online]. Available: http://dx.doi.org/10.1137/S0895479898346995

[61] S. Weiland and F. van Belzen, "Singular value decompositions and low rank approximations of tensors," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1171–1182, March 2010.

[62] A. P. da Silva, P. Comon, and A. L. F. de Almeida, "Rank-1 tensor approximation methods and application to deflation," *arXiv preprint arXiv:1508.05273*, 2015.

[63] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009. [Online]. Available: http://dx.doi.org/10.1137/07070111X

[64] L. D. Lathauwer, "A survey of tensor methods," in *2009 IEEE International Symposium on Circuits and Systems*, May 2009, pp. 2773–2776.

[65] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.

[66] P.-Ã. Wedin, "Perturbation bounds in connection with singular value decomposition," *BIT Numerical Mathematics*, vol. 12, no. 1, pp. 99–111, 1972.

[67] D. A. Abraham and N. L. Owsley, "Beamforming with dominant mode rejection," in *Conference Proceedings on Engineering in the Ocean Environment*, Sep 1990, pp. 470–475.

[68] R. T. Suryaprakash, B. E. Moore, and R. R. Nadakuditi, "Algorithms and performance analysis for estimation of low-rank matrices with kronecker structured singular vectors," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 3776–3780.

[69] C.-Y. Chen and P. Vaidyanathan, "A subspace method for mimo radar space-time adaptive processing," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 2, April 2007, pp. II–925–II–928.

[70] A. Haimovich, "The eigencanceler: Adaptive radar by eigenanalysis methods," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 32, no. 2, pp. 532–542, 1996.

[71] J. R. Buck and K. E. Wage, "A random matrix theory model for the dominant mode rejection beamformer notch depth," in *2012 IEEE Statistical Signal Processing Workshop (SSP)*, Aug 2012, pp. 820–823.

[72] A. Nehorai and E. Paldi, "Acoustic vector-sensor array processing," *IEEE Transactions on Signal Processing*, vol. 42, no. 9, pp. 2481–2491, Sep 1994.

[73] M. Hawkes and A. Nehorai, "Acoustic vector-sensor correlations in ambient noise," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 3, pp. 337–347, Jul 2001.

[74] J. Li, P. Stoica, and Z. Wang, "On robust capon beamforming and diagonal loading," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1702–1715, July 2003.

[75] D. R. Fuhrmann, "Application of toeplitz covariance estimation to adaptive beamforming and detection," *IEEE Transactions on Signal Processing*, vol. 39, no. 10, pp. 2194–2198, Oct 1991.

[76] D. D. Feldman and L. J. Griffiths, "A projection approach for robust adaptive beamforming," *IEEE Transactions on Signal Processing*, vol. 42, no. 4, pp. 867–876, Apr 1994.

[77] J. R. Guerci, "Theory and application of covariance matrix tapers for robust adaptive beamforming," *IEEE Transactions on Signal Processing*, vol. 47, no. 4, pp. 977–985, Apr 1999.

[78] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero, "Shrinkage algorithms for mmse covariance estimation," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5016–5029, Oct 2010.

[79] L. L. Scharf, "The svd and reduced rank signal processing," *Signal Processing*, vol. 25, no. 2, pp. 113 – 133, 1991. [Online]. Available: http://www.sciencedirect.com/science/article/pii/016516849190058Q

[80] K. E. Wage and J. R. Buck, "Snapshot performance of the dominant mode rejection beamformer," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 2, pp. 212–225, April 2014.

[81] V. A. Marčenko and L. A. Pastur, "Distribution of eigenvalues for some sets of random matrices," *Sbornik: Mathematics*, vol. 1, no. 4, pp. 457–483, 1967.

[82] M. A. O. Vasilescu and D. Terzopoulos, *Multilinear (Tensor) ICA and Dimensionality Reduction*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 818–826.

[83] J. F. Cardoso, "Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem," in *International Conference on Acoustics, Speech, and Signal Processing*, Apr 1990, pp. 2655–2658 vol.5.

[84] Y. Li and A. Ngom, "Non-negative matrix and tensor factorization based classification of clinical microarray gene expression data," in *Bioinformatics and Biomedicine (BIBM), 2010 IEEE International Conference on*. IEEE, 2010, pp. 438–443.

[85] E. Acar, C. Aykut-Bingol, H. Bingol, R. Bro, and B. Yener, "Multiway analysis of epilepsy tensors," *Bioinformatics*, vol. 23, no. 13, pp. i10–i18, 2007.

[86] A. Anandkumar, R. Ge, D. J. Hsu, S. M. Kakade, and M. Telgarsky, "Tensor decompositions for learning latent variable models." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2773–2832, 2014.

[87] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, 2013. [Online]. Available: http://dx.doi.org/10.1002/gamm.201310004

[88] D. Goldfarb and Z. T. Qin, "Robust low-rank tensor recovery: Models and algorithms," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 1, pp. 225–253, 2014. [Online]. Available: http://dx.doi.org/10.1137/130905010

[89] J. Chen and Y. Saad, "On the tensor svd and the optimal low rank orthogonal approximation of tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 4, pp. 1709–1734, 2009. [Online]. Available: http://dx.doi.org/10.1137/070711621

[90] L. Eldén and B. Savas, "A newtonÃćÂĂÂŞgrassmann method for computing the best multilinear rank-$(r_1,$ $r_2,$ $r_3)$ approximation of a tensor," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 2, pp. 248–271, 2009. [Online]. Available: http://dx.doi.org/10.1137/070688316

[91] T. Zhang and G. H. Golub, "Rank-one approximation to high order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 2, pp. 534–550, 2001. [Online]. Available: http://dx.doi.org/10.1137/S0895479899352045

[92] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, p. 025010, 2011. [Online]. Available: http://stacks.iop.org/0266-5611/27/i=2/a=025010

[93] A. Stegeman and N. D. Sidiropoulos, "On kruskals uniqueness condition for the candecomp/parafac decomposition," *Linear Algebra and its Applications*, vol. 420, pp. 540 – 552, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0024379506003855

[94] N. K. M. Faber, R. Bro, and P. K. Hopke, "Recent developments in candecomp/parafac algorithms: a critical review," *Chemometrics and Intelligent Laboratory Systems*, vol. 65, no. 1, pp. 119 – 137, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169743902000898

[95] E. Acar, D. M. Dunlavy, and T. G. Kolda, "A scalable optimization approach for fitting canonical tensor decompositions," *Journal of Chemometrics*, vol. 25, no. 2, pp. 67–86, February 2011.

[96] B. W. Bader, T. G. Kolda *et al.*, "Matlab tensor toolbox version 2.6," Available online, February 2015. [Online]. Available: http://www.sandia.gov/~tgkolda/TensorToolbox/

[97] S. Liu, "Matrix results on the khatri-rao and tracy-singh products," *Linear Algebra and its Applications*, vol. 289, no. 1-3, pp. 267–277, 1999.

[98] S. Liu and G. Trenkler, "Hadamard, khatri-rao, kronecker and other matrix products," *Int. J. Inf. Syst. Sci*, vol. 4, no. 1, pp. 160–177, 2008.

[99] K. B. Petersen, M. S. Pedersen *et al.*, "The matrix cookbook," *Technical University of Denmark*, vol. 7, p. 15, 2008.

[100] Z. Bai and J. Silverstein, *Spectral analysis of large dimensional random matrices*. Springer Verlag, 2010.

[101] R. Latała, "Some estimates of norms of random matrices," *Proceedings of the American Mathematical Society*, vol. 133, no. 5, pp. 1273–1282, 2005.

[102] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2010.

[103] T. Tao, *Topics in random matrix theory*. AMS Bookstore, 2012, vol. 132.

[104] L. Pastur and V. Vasilchuk, "On the law of addition of random matrices," *Communications in Mathematical Physics*, vol. 214, no. 2, pp. 249–286, 2000.

[105] W. Hachem, P. Loubaton, J. Najim, P. Vallet *et al.*, "On bilinear forms based on the resolvent of large random matrices," in *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, vol. 49, no. 1. Institut Henri Poincaré, 2013, pp. 36–63.