

**HIGH FIDELITY DECONVOLUTION
OF MODERATELY RESOLVED SOURCES**

by

DANIEL SHENON BRIGGS, B.S., B.S., M.S.

Submitted in Partial Fulfillment
of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY IN PHYSICS

The New Mexico Institute of Mining and Technology
Socorro, New Mexico

March, 1995

To my Mother and the memory of my Father

ABSTRACT

This dissertation contains several topics related to high fidelity imaging with interferometers, including deconvolution simulations to show quantitatively how well existing algorithms do on simple sources, a new deconvolution algorithm which works exceedingly well but can only be applied to small objects, and a new weighting scheme which offers mild improvement to nearly any observation.

Robust weighting is a new form of visibility weighting that varies smoothly from natural to uniform weighting as a function of a single real parameter, the robustness. Intermediate values of the robustness can produce images with moderately improved thermal noise characteristics compared to uniform weighting at very little cost in resolution. Alternatively, an image can be produced with nearly the sensitivity of the naturally weighted map, and resolution intermediate between that of uniform and natural weighting. This latter weighting often produces extremely low sidelobes and a particularly good match between the dirty beam and its fitted Gaussian, making it an excellent choice for imaging faint extended emission.

A new deconvolver has been developed which greatly outperforms CLEAN or Maximum Entropy on compact sources. It is based on a preexisting Non-Negative Least Squares matrix inversion algorithm. NNLS deconvolution is somewhat slower than existing algorithms for slightly resolved sources, and very much slower for extended objects. The solution degrades with increasing source size and at the present computational limit (~ 6000 pixels of significant emission) it is roughly comparable in deconvolution fidelity to existing algorithms. NNLS deconvolution is particularly well suited for use in the self-calibration loop, and for that reason may prove particularly useful for Very Long Baseline Interferometry, even on size scales where it is no better than existing deconvolvers.

The basic practice of radio interferometric imaging was re-examined to determine fundamental limits to the highest quality images. As telescopes have become better, techniques which served an earlier generation are no longer adequate in some cases. Contrary to established belief, the deconvolution process itself can now contribute an error comparable to that of residual calibration errors. This is true for even the simplest imaging problems, and only the fact that the error morphology of deconvolution and calibration errors are similar has masked this contribution until now. In cases where it can be applied, these deconvolution problems are largely cured by the new NNLS deconvolver. An extensive suite of simulations has been performed to quantify the expected magnitude of these errors in a typical observation situation.

The new techniques have been demonstrated by observational projects with the Very Large Array, Australia Telescope Compact Array and Very Long Baseline Array on the sources 3C48, SN1987A and DA193 respectively. The 3C48 project was designed to trace or exclude extended emission from a VLBI scale disrupted jet, and yielded a null result at a noise limited dynamic range of 180,000:1 from an extended object. The SN1987A project was designed for the highest resolution imaging possible and yielded high confidence level astrophysically important structure at half the synthesized uniform beamwidth. The DA193 project was primarily a test of the new VLBA telescope, but yielded as a by product the highest dynamic range images ever produced by VLBI. There are no comparable observations on other telescopes for comparison, but the observed 115,000:1 exceeded the previous record by more than a factor of 10.

Acknowledgments

When first I presented a topic to my dissertation committee, it was a somewhat nebulous collection of problems that I wanted to address, mostly in the area of developing sophisticated new calibration techniques for radio interferometers. Unfortunately, while I had initial strategies for these investigations, I lacked a clear vision of which problems would prove most tractable or most useful. The proposal was based on the theory that with enough promising beginnings, at least some were likely to succeed. My committee gifted me with a great deal of freedom in my pursuits, for which vote of confidence I am profoundly grateful. That freedom allowed me to follow my nose to this dissertation, whose results have surprised us all. There is little of the original topic left; the curious behavior encountered in the initial “warm up” deconvolution project grew so intriguing as to occupy nearly all of my time. The subject of calibration is now touched on only peripherally in the pages that follow.

There are many, many people who deserve thanks and acknowledgements in a work like this. But first among them I wish to thank Tim Cornwell, my scientific advisor. I have been tremendously lucky to encounter an advisor with whom I was so professionally and personally compatible. I could not ask for an advisor more skilled in his art and generous in his time in explaining it. And while always delivered with compassion, I greatly appreciate the honesty and candor of his progress assessments — both good and bad. It is difficult to properly express the debt I feel. My fond hope is that I come to reflect well on him as I continue to grow and mature as a scientist.

I have actually been very lucky with *all* of the advisors and committee members that I have come in contact with during my graduate career. Jean Eilek has seen me through many years as my academic advisor, and taught me much of what I know about astrophysics. I’m sorry it worked out that she could not be present for the final defense; I would have liked her signature on my dissertation. A friend as well as teacher, she is one of the rare people on the planet who can out talk me by 50% on nearly any subject of mutual interest. (Pity the poor observer trying to follow an excited discussion between us!) Alan Blyth has done an admirable job of stepping into her place as academic advisor and looking after my interests while Jean is on sabbatical. Craig Walker also stands out in my mind as a particularly significant mentor and companion during the past years. His unflagging good humor and superb technical competence has been greatly appreciated as we worked on a number of projects together, both my master’s degree project and the 3C48 project which inspired much of the work in this dissertation among them. I’d like to thank Joan Wrobel for making me welcome at NMT/NRAO and involving me in her research almost from the moment I arrived in Socorro. In the several years she advised me during my Master’s degree I have always been struck by her desire to do right by the student. It

has been much appreciated! Tim Hankins is another committee member who could not participate in the final defense due to a sabbatical at extreme distance. His presence was missed and I thank both Dave Westpfahl and Paul Krehbiel for stepping in to fill the empty positions under somewhat difficult circumstances.

I also particularly wish to thank my supervisor-to-be, Dave Mozurkewich, for graciously allowing me to defer the start of my first postdoc and complete this dissertation substantially to my satisfaction. There are certainly rough spots, and more work could be done in a number of areas — but several important lines of research paid off dramatically right at the end of my NRAO fellowship and without the cooperation of both Dave and my supervisors at NRAO, much useful material would not be present in this document. I am grateful for the extra time and use of NRAO facilities during the awkward transition, and the dissertation is much the better for it.

I have been present at the Array Operations Center in Socorro during the final construction years and dedication of the Very Long Baseline Array. The VLBA is a superb new instrument, and everyone associated with it deserves the praise of the astronomical community. But as I've worked (and played) through many long nights at the AOC, I've personally seen the heroic efforts of Craig Walker and of Jon Romney in bringing the VLBA to life. The last image of DA193 in Chapter 9 is more than an order of magnitude deeper in dynamic range than any other VLBI image of which I am aware. This image is a testament to the dedication and skill of everyone who worked on VLBA. But in terms of the human sacrifice necessary to achieve such incredible instrumental calibration, stability and correlator performance, I will always associate that with Craig and Jon.

New Mexico Tech happens to require a considerable amount of classwork of its physics graduate students, and consequently I've enjoyed many classes from some first rate people. Jean, of course, Bob Hjellming, Tim Hankins and Albert Petschek. Thanks to you all; I'm sure it'll keep me in good stead in the future. And special thanks to Rick Perley not only for an excellent class in Fourier theory, but also for steering me towards Tim Cornwell as an advisor.

The list of “helpful discussions” reads much like the scientific staff phone list: Dave Adler, Durga Bagri, Claire Chandler, Barry Clark, Richard Davis, Phil Diamond, Dwaraka, Chris Flatters, Ed Fomalont, Dale Frail, Jim Higdon, Mark Holdaway, Ralph Marson, Frasier Owen, Peter Napier, Michael Rupen, Bob Sault, Fred Schwab, Ulrich Schwarz, Ken Sowinski, Dick Sramek, Lister Stavely-Smith, Tom Tongue, Juan Uson, Dave Westpfahl, Doug Woods, Gustaaf van Moorsel and Anton Zensus. Some of these people I've worked with fairly closely. Others have suffered through being test cases of my experimental software or read drafts for me. Still others have offered useful comments over coffee. All of them have given me something, and I thank them for it.

Ruth Milner, Gustaaf van Moorsel, Wes Young and Jon Spargo have impressed me particularly as the heart of the computer division here at the AOC. Working under difficult

budget and manpower limitations, they never loose sight that the *purpose* of computers is to help people solve problems. As one of the more computer intensive individuals in the building I've occasionally managed to dig myself into some real holes, and they've always bailed me out.

Harry Andrews & Bobby Hunt deserve mention, even though I have met Bobby but briefly and Harry not at all. Their book, *Digital Image Restoration* (Andrews and Hunt, 1977) is simply amazing, and with the benefit of 18 years hindsight seems nothing short of prescient. If it had been written at a time with the computer resources of today, I might have had little left to do in Chapter 4.

I spent the first 5 months of 1993 in Australia, as the guest of the Australia Telescope National Facility. I enjoyed my time there tremendously, and it was quite productive scientifically. Chapter 8 is almost entirely based on work done there. I also wrote the NNLS algebraic deconvolution program while at the ATNF, but did not recognize at the time how useful it would become. Ron Ekers, Neil Killeen, Bob Sault and Ralph Marson all made me feel very welcome. Lister Stavely-Smith did as well, and graciously allowed me to work on his lovely SN1987A data. Also while down under, I fortunate to form a marvelous friendship with Sue Byleveld.

On the personal side, thanks to Michael Rupen for many things: the incredibly detailed dissertation comments went far above and beyond the call of duty, and this document is much the better for the help. But even more than that, thanks for the friendship and moral support during some pretty bleak times when finishing up. Thanks to John Biretta for the milk and cookies that appeared without fail at every late night Star Trek session, to Jimbo Higdon for the Nerf wars, to Dave Moffett for the movie nights at the bomb shelter and all too infrequent rock climbs. To Tom & Maggie Tongue, Andrew Lobanov, Kari Leppanen, Chris Taylor, Beth Hufnagel, Kevin Marvel, Chris de Pree and Lisa Young as comrades in arms during the long graduate struggle.

As to personal friends beyond work, I could ask for none better than Kathy Hedges. Whether it's been out square dancing, an evening's quiet tea, or being her 'best man', I've enjoyed it all! To Peggi & Kevin Whitmore and all the other nifty people I've met through English, Contra and Morris dancing — I've had a great time. Keep dancing, and I'll be back to join you occasionally. And no discussion of my life as a graduate student would be complete without mentioning the joy I've found skydiving. The people I've met at the drop zone have been lovely: John Walter, Cynthia Cooper, Mike & Jane Ann Bode and many many others. Thanks to all my skydiving friends for flying with me, keeping me sane, and teaching me how to know what is *really* important.

And of course, with all my I heart I thank my parents to whom this dissertation is dedicated. I'm sorry my father could not have lived to see it, and I miss him very much.

Financial support for the research in this dissertation has come from an NRAO Junior Research Fellowship, from a New Mexico Tech Research Assistantship, and from the Australia Telescope National Facility.

Finally, except where stated otherwise, all work in this dissertation is original. I gratefully acknowledge the initial concept of Robust Weighting to Tim Cornwell, and the initial concept of beam forcing though a Gaussian taper to Bill Cotton. Rick Perley first pointed out to me that the CLEAN algorithm could be solved analytically in the limit of perfect u - v sampling and critical sampling, and Craig Walker suggested successive differencing as a superior alternative to boxcar averaging in the estimation of thermal noise.

DANIEL SHENON BRIGGS

The New Mexico Institute of Mining and Technology

March, 1995

Table of Contents

ABSTRACT	v
Acknowledgments	vii
Table of Contents	xi
List of Figures	xvii
List of Tables	xxiii
List of Programs	xxv
List of Symbols and Functions	xxvii
1. Introduction	1
1.1 Overview of Dissertation Structure	4
2. Representation of the Deconvolution Equation	7
2.1 The Visibility Equations	7
2.2 The Continuous Dirty Map	13
2.3 Discretization of the Continuous Equations	14
2.4 Convolutional Gridding and Aliasing	17
2.5 Limitations of the even-sized FFT	18
2.5.1 Non-Power-of-2 FFT Performance Analysis	19
2.6 Deconvolution	24
2.7 Deconvolution Algorithms	27
2.7.1 CLEAN	27
2.7.2 SDI CLEAN	29

2.7.3	Multi Resolution CLEAN	30
2.7.4	Maximum Entropy	30
2.7.5	Non-Negative Least Squares	31
2.7.6	Maximum Emptiness	32
2.7.7	Gerchberg-Saxon-Papoulis	32
2.7.8	Richardson-Lucy	33
2.7.9	Singular Value Decomposition	33
3.	Robust Weighting	35
3.1	Introduction	35
3.2	Thermal noise in the synthesized map	36
3.3	Weighting as the Solution to a Minimization Problem	40
3.3.1	Natural Weighting	40
3.3.2	Sensitivity to Resolved Sources	41
3.3.3	Uniform Weighting	43
3.3.4	Super-Uniform Weighting	45
3.4	Beamwidth and Thermal Noise	47
3.5	Catastrophic Gridding Errors	49
3.5.1	Gridless Weighting	55
3.6	Robust Uniform Weighting	57
3.6.1	Robust Weighting and Wiener Filtering	59
3.6.2	Specification of the Robustness	61
3.6.3	Example Thermal Noise Calculations	63
3.7	Simulations	72
3.7.1	VLA Full Tracks	74
3.7.2	VLA Full Tracks (3C48)	75
3.7.3	VLA 30 Point Snapshot	75
3.7.4	VLA 5 Point Snapshot	77

3.7.5	VLA Multiple Configurations (M82)	78
3.7.6	AT Full Tracks (SN1987A)	81
3.7.7	VLBA Full Tracks	81
3.7.8	VLBA/VLA/GBT Full Tracks	82
3.7.9	VLBA/VLA/GBT/Orbiter	84
3.8	RMS/resolution Tradeoff Curves	108
3.8.1	Tradeoff Parameterized by Robustness	108
3.8.2	Thresholding	111
3.8.3	Time Averaging	113
3.8.4	Super-Uniform Weighting	116
3.9	Automatic selection of Robustness	124
3.10	Model Deconvolutions	126
3.11	Tapering & Beam Forcing	134
3.11.1	VLB (Square Root) Weighting	155
3.12	Future Work	162
3.12.1	Azimuthally Varying Robustness	162
3.12.2	Thermal Noise Beam and Visualization	162
3.12.3	Generalized Windows and Minimizing Source Sidelobes	164
3.13	Summary	166
3.13.1	Imaging Recommendations	166
3.13.2	Software Recommendations	167
4.	Algebraic Deconvolution	169
4.1	Representation	171
4.2	Singular Value Decomposition	177
4.3	NNLS	181
4.3.1	The NNLS algorithm	182
4.3.2	How Important are the Windows?	184

4.3.3	Model Deconvolutions	184
4.3.4	Discussion	196
5.	Simulations	199
5.1	The Resolution Test Deconvolutions	200
5.2	Error Morphology	203
5.2.1	Results	229
5.3	Residual Behavior	231
5.4	Algorithmic Cross Comparison	233
5.5	The Effect of Fourier Plane Coverage	241
5.6	Flux estimation	246
5.7	Gaussian Pixel Functions	251
5.8	Point Sources Between Pixels	256
5.9	Convergence in the Fourier Plane	262
5.10	3C48 Simulations — MEM	265
6.	Miscellaneous Topics	269
6.1	Estimation of Thermal Noise from the Visibility Data	269
6.2	CLEAN component placement	272
6.2.1	The Gaussian approximation to initial CLEANing	272
6.2.2	The Gaussian Approximation Applied to a VLA PSF	277
6.2.3	Multiple Orders of Clean Components	280
6.2.4	Fitting to CLEANed Gaussians	283
6.3	CLEAN components are Fourier Components in the Limit	287
6.4	Model Combination in the Fourier and Image Planes	289
7.	Case Study: 3C48	295
7.1	Introduction	295
7.2	The Data and the Noise	296

7.3	Modelling	299
7.4	X Band Images	302
7.5	L Band Images	307
7.6	Discussion	312
8.	Case Study: SN1987A	313
8.1	Introduction	313
8.2	The Data	314
8.3	Deconvolution	315
8.3.1	Support Constraints	321
8.3.2	Cross Validation	322
8.4	Algorithmic Convergence	323
8.4.1	Linear Summation	326
8.4.2	Restored Images	327
8.4.3	Comparison with Other Data	330
8.4.4	Summary	332
8.5	Visibility Weighting and Thermal Noise	333
8.6	Modelling	339
8.7	Results and Discussion	344
9.	Case Study: DA193	347
9.1	VLB Array Memo #697A	349
9.1.1	Introduction	349
9.1.2	Initial Calibration	349
9.1.3	CLEAN Deconvolution	350
9.1.4	Imaging Presentation and Metrics	352
9.1.5	Model Fit with CLEAN Tidying	352
9.1.6	NNLS	354

9.1.7	Noise Comparison	357
9.1.8	Super Resolution	358
9.1.9	Self-Calibration	361
9.1.10	Conclusions and Further Work	361
9.2	Simulations	362
9.2.1	Gaussian Model	362
9.2.2	Gain Transfer	364
9.2.3	Halo Model	368
9.3	Conclusions and Further Work	373
10.	Concluding Comments	375
A.	Lagrange Multipliers and the Weighted Mean	379
B.	Elliptical Gaussian Convolution	381
C.	Beam Fitting	385
D.	Generalized Tapers and Beam Forcing	387
E.	Gridless Weighting Algorithm	390
E.1	Variations of the Algorithm	394
E.2	Automatic Bin Size Selection	397
F.	Complex Fourier Series	402
G.	SDE	403
H.	Electronic Supplement	404
Bibliography		405

List of Figures

2.1	Radiation on the Celestial Sphere	9
2.2	Normalized FFT Runtimes vs. Size	20
3.1	Resolution vs. Thermal Noise Tradeoff	50
3.2	Catastrophic Gridding Error in the u - v Plane	51
3.3	Striping in the Dirty Map	53
3.4	Striping in the Restored Image	54
3.5	Isolated u - v Sampling Points	56
3.6	Thermal Noise-Resolution Tradeoff — Robustness	63
3.7	Thermal Noise-Resolution Tradeoff — Robustness with eccentricity.	64
3.8	Azimuthally Averaged Gridded Total Weights — VLA Full Tracks	65
3.9	Azimuthally Averaged Gridded Density Weights — VLA Full Tracks	66
3.10	Density Weight Histograms — VLA Full Tracks	67
3.11	Model Density Weight Histograms	69
3.12	Fourier Plane Coverages	86
3.13	PSF Plots vs. Robustness, VLA Full Tracks	88
3.14	PSF Plots vs. Robustness, VLA Full Tracks (3C48)	90
3.15	PSF Plots vs. Robustness, VLA 30 Point Snapshot	92
3.16	PSF Plots vs. Robustness, VLA 30 Point Snapshot, $FOV_w = 1/4$	94
3.17	PSF Plots vs. Robustness, VLA 5 Point Snapshot, $FOV_w = 1/4$	96
3.18	PSF Plots vs. Robustness, VLA Multiple Config (M82)	98
3.19	PSF Plots vs. Robustness, AT Full Tracks (SN1987A)	100
3.20	PSF Plots vs. Robustness, VLBA Full Tracks	102
3.21	PSF Plots vs. Robustness, VLBA/VLA/GBT Full Tracks	104
3.22	PSF Plots vs. Robustness, VLBA/VLA/GBT/Orbiter	106
3.23	Robustness Tradeoff Curves	109
3.24	Thresholding Tradeoff Curve, VLA Full Tracks	111
3.25	Time-averaging Tradeoff Curves	114

3.26	RMS vs. Beamsize, varying FOV_w	118
3.27	PSF Plots vs. FOV, VLA Full Tracks (3C48)	120
3.28	Super-Uniform Weighting as a Way to Prevent a Shelf	123
3.29	Automatic Selection of Robustness by Gaussian Best-Fit	125
3.30	3C48 Model Deconvolutions — CLEAN	128
3.31	Error Quantities against Thermal RMS — CLEAN	131
3.32	Beam Circularization by Tapering	135
3.33	Beam Slices Against Simple Tapering	136
3.34	Robustness and Tapering	141
3.35	PSF Plots vs. Robustness, Tapered, $FOV_w = 1.0$	148
3.36	PSF Plots vs. Robustness, Tapered, $FOV_w = 0.25$	150
3.37	PSFs Tapered to Fixed Resolution	153
3.38	Robustness Tradeoff Curves, VLBI Weighting Comparison.	157
3.39	PSF Plots vs. Robustness, VLBA/VLA/GBT Full Tracks, VLB weighting	158
3.40	PSF Plots vs. Robustness, VLBA/VLA/GBT/Orbiter, VLB weighting	160
3.41	Azimuthally Varying Robustness — AT Full Tracks (SN1987A)	163
4.1	Singular Value Decomposition of a One-Dimensional PSF	174
4.2	Two-Dimensional Beam Matrix	176
4.3	Singular Value Spectrum vs. Window Size	178
4.4	SVD Deconvolutions of 3C48 Model Source	180
4.5	3C48 Raw and Smoothed Model Source	186
4.6	CLEAN Deconvolution	187
4.7	NNLS Deconvolution with Faint Sources Present	188
4.8	NNLS Deconvolution without Faint Sources	189
4.9	Error Quantities against Thermal RMS — NNLS	191
4.10	Missing Data Deconvolution	195
5.1	Error Characterization Simulations, CLEAN, Gaussian Model	204
5.2	Error Characterization Simulations, MEM, Gaussian Model	208
5.3	Error Characterization Simulations, NNLS, Gaussian Model	212
5.4	Error Characterization Simulations, CLEAN, Disk Model	216

5.5	Error Characterization Simulations, MEM, Disk Model	220
5.6	Error Characterization Simulations, NNLS, Disk Model	224
5.7	Model Error Amplitude in the Fourier Plane — Algorithmic Comparison . .	228
5.8	Image Plane Residual RMS	232
5.9	Deconvolution Errors vs. Source Size, Algorithmic Comparison	234
5.10	Deconvolution Errors vs. Reconvolution Size, Algorithmic Comparison . .	237
5.11	Deconvolution Errors vs. Source Size, Scaled 3C48 Model.	240
5.12	Mid Tracks Fourier Plane Coverage and PSF	242
5.13	Perfect VLA Fourier Plane Coverage and PSF	242
5.14	Fourier Plane Coverage and Deconvolution Errors	243
5.15	Flux Estimation Errors	247
5.16	Smooth Pixel CLEAN Deconvolution Errors vs. Source Size	252
5.17	Smooth Pixel CLEAN Deconvolution Errors vs. Reconvolution Size	254
5.18	Shifted Point Source — Fourier Plane Coverage and PSF	257
5.19	Shifted Point Source Images	258
5.20	Shifted Point Source — Fourier Plane Convergence	263
5.21	Error Quantities against Thermal RMS — MEM	266
6.1	Creation of Local Maxima in Smooth Emission	273
6.2	Example of Initial CLEANing	274
6.3	CLEAN Component Images, Gaussian Beam	276
6.4	CLEAN Component Sequence	276
6.5	CLEAN Component Radial Distance vs. Iteration	277
6.6	CLEAN Component Images, VLA Beam	278
6.7	CLEAN Component Radial Distance vs. Iteration, VLA PSF	279
6.8	Convolution Behavior of the VLA PSF	279
6.9	CLEAN Component Radial Distance vs. Iteration	282
6.10	CLEANed Gaussian Model Fits	284
6.11	Model Combination in the Fourier Plane	291
6.12	CLEAN + MEM Sequentially in the Image Plane	293
7.1	Visibility Amplitude Plot	296

7.2	Diagnosis of Deconvolution Problems by Modelling	301
7.3	3C48 X-Band Images, Various Weightings and Deconvolvers	303
7.4	Visibility Amplitude Plot	307
7.5	Wide-Field L-Band Naturally Weighted Image	310
8.1	Visibility Amplitude Plot	314
8.2	Nominal Images for Different Deconvolution Algorithms	316
8.3	Sky Models for Different Deconvolution Algorithms	319
8.4	Windowed and Unwindowed MEM Deconvolution	321
8.5	Cross Validation	322
8.6	MEM Convergence	324
8.7	Over-Iteration	325
8.8	Linear Combination of Subset Images	326
8.9	Super-Resolved Images vs. Algorithm	329
8.10	Comparison with other Data	331
8.11	PSF Slices vs. Weighting	334
8.12	Sky Models vs. Weighting	337
8.13	Modelling of Spurious Peaks	342
8.14	Modelling of PSF Position Angle Bias	343
9.1	Observational and Imaging Parameters	348
9.2	CLEAN Images of Two Different IFs	351
9.3	Model-Fitting + CLEAN, NNLS Images	353
9.4	Best Uniform and Naturally Weighted NNLS Images	356
9.5	Super-Resolved Images	360
9.6	Gaussian Model with Thermal Noise	363
9.7	Correlated Gain Example — Gaussian Model	364
9.8	Offsource RMS vs. Self-Cal Iteration	365
9.9	Deconvolutions of Self-Calibrated Gaussian Model Data	367
9.10	Correlated Gain Example — NNLS Model	369
9.11	Halo Model and Thermal Noise	370
9.12	Deconvolutions of Self-Calibrated Halo Model Data	372

9.13 8 IF High Dynamic Range Image	374
E.1 Geometry of the Binning Array	394

List of Tables

2.1	Odd FFT Sizes Between 33 and 2187 with Efficiency Loss Better than 25%	23
2.2	All FFT Sizes Between 32 and 2187 with Efficiency Loss Better than 25% .	23
3.1	Uniform Beamwidth vs. “Innocuous” Imaging Parameters	48
3.2	Natural Beamwidth vs. “Innocuous” Imaging Parameters	49
3.3	Thermal Noise vs. “Innocuous” Imaging Parameters, Uniform Weighting .	49
3.4	Analytical RMS for Case A	70
3.5	Analytical RMS for Case B	71
3.6	Analytical RMS for Case C	72
3.7	Simulation Parameters	74
3.8	3C48 & Point Source Model Parameters	126
3.9	Gridded/Gridless RMS Comparison	139
3.10	Robustness/Taper Tradeoff, RMS Comparison	144
3.11	Robustness/Taper Tradeoff, Negative Sidelobe Comparison	145
E.1	Run Times for the Gridless Algorithm Main Loop.	400
E.2	Run Time Excesses Using Optimum Binning Array Size Approximation .	401

List of Programs

B.1	Conversion of Quadratic Parameters to Elliptical/Hyperbolic Parameters	384
D.1	Beam Forcing Iteration	389
E.1	Visibility Binning Procedure	392
E.2	Simple Gridless Weighting Procedure	393
E.3	Central Decision Tree for Quick Gridless Weighting	395

List of Symbols and Functions

Symbols

λ	wavelength
ν	frequency
u, v	antenna spacing coordinates in units of wavelength
w	antenna spacing coordinate in units of wavelength, weighting function
ℓ, m	direction cosines on the sky conjugate to u and v
u_{max}, v_{max}	maximum values of u and v present in a data set
$\hat{u}_{max}, \hat{v}_{max}$	maximum values of u and v representable on the gridded u - v plane
ℓ_{max}, m_{max}	maximum values of ℓ and m representable in the image
$\Delta\ell, \Delta m$	pixel spacing on the sky along the ℓ and m axes
$\Delta u, \Delta v$	pixel spacing in the Fourier plane along the u and v axes
Δv	pixel spacing on the Fourier plane along the v axis
$I(\ell)$	true sky intensity
$I^D(\ell)$	dirty map of the sky
$B(\ell)$	dirty beam, also called the point spread function
$C(\ell)$	component model of the sky
$P(\ell)$	pixel model or pixel function
\mathbf{t}	true sky distribution vector
\mathbf{d}	dirty sky distribution vector
\mathbf{r}	deconvolution residual vector
\mathbf{c}	component representation vector
\mathcal{V}_k	true sky visibility at spacing (u_k, v_k)
V_k	measured sky visibility at spacing (u_k, v_k)
D_k	density weight for visibility k
T_k	taper for visibility k
w_k	signal-to-noise weight for visibility k
W_k	total weight $\equiv T_k D_k w_k$
D_{pq}	density weight for gridded visibility $p q$
T_{pq}	taper for gridded visibility $p q$
w_{pq}	signal-to-noise weight for gridded visibility $p q$
W_{pq}	total gridded weight $\equiv T_{pq} D_{pq} w_{pq}$
FOV_w	weighting field of view. $FOV_w = 1 \Rightarrow$ uniform weighting
RMS	root mean square
$k \in c_{pq}$	all indices k such that (u_k, v_k) is in gridding cell (p, q)

Functions

$\operatorname{Re} \mathcal{V}$	Real part of \mathcal{V}
$\operatorname{Im} \mathcal{V}$	Imaginary part of \mathcal{V}
$\operatorname{sinc}(x)$	$\sin(\pi x)/(\pi x)$
$\delta(x)$	Dirac delta function
z^*	complex conjugate of z
$a * b$	convolution of functions a and b
$\langle e \rangle$	expectation of expression e
$O(f(x))$	order of: the leading behavior of $f(x)$ as $x \rightarrow \infty$
$e_1 \vee e_2$	Boolean logical OR of expressions e_1 and e_2
$e_1 \wedge e_2$	Boolean logical AND of expressions e_1 and e_2
$[x]$	ceiling, greatest integer $\leq x$
$\lfloor x \rfloor$	floor, least integer $\geq x$
$\ \mathbf{x}\ _n$	L_n norm of the vector \mathbf{x} , $\left(\sum_k x_k ^n \right)^{1/n}$ $\ \mathbf{x}\ \equiv \ \mathbf{x}\ _2$
$\text{III}(x)$	shah function, $\sum_{i=-\infty}^{\infty} \delta(x - i)$
$\Pi(x)$	rectangle function, $\begin{cases} 1 & x < 1/2 \\ 1/2 & x = 1/2 \\ 0 & \text{otherwise} \end{cases}$

Fourier Transforms

$\mathfrak{F}[f]$	Fourier transform of function f , see below
$\mathfrak{F}^{-1}[F]$	inverse Fourier transform of function F
$\mathfrak{F}[F(u)] \rightleftharpoons \mathfrak{F}^{-1}[f(\ell)]$	$F(u)$ transforms to $f(\ell)$

In the “transforms to” symbol, the signs refer to that of the transform kernel. The forward Fourier transform, the “minus i ” direction, is defined as

$$\mathfrak{F}[f] = \int_{-\infty}^{\infty} f(\ell) e^{-i2\pi u\ell} d\ell$$

Generic Notation

\mathbf{r}	spatial vector are boldface, and may be upper or lower case roman letters
\mathbf{A}	matrices are upper case roman letters
\mathbf{x}	arbitrary vectors are lower case roman letters
a_k	k indexes over an ungridded quantity
a_{pq}	pq indexes over a gridded quantity

Chapter 1

Introduction

Images are the basic input to spatially resolved studies of nearly everything in astronomy. In some contexts the problem of imaging is intertwined with additional considerations such as spectroscopy, but in nearly every spatially resolved application there comes a point where one is interested in making the best images possible given the raw data.

The very concept of “best image” is not easily quantifiable, and will depend on the application. One might be interested in seeing as much detail in the image as possible — provided that the detail seen can be believed. One might be interested in minimizing the cross contamination between different objects in the same field, or in recovering a reliable total flux in the presence of noise, or in maximizing the sensitivity of the images while preserving other desirable qualities so far as possible. Or one might even simply desire the most aesthetic image possible, free of obvious defects and perhaps distorted in some known way as the easiest image for general morphological understanding. All of this is hampered by the fact that we generally do not have the true image for comparison, but there are consistency checks which can be performed to increase the likelihood of telling a good image from a bad one.

This dissertation addresses a diverse range of topics within astronomical interferometric imaging, all of which can be loosely grouped together under the topic of “High Fidelity Deconvolution.” A convolution equation is of the form

$$f(x) * g(x) \equiv \int_{-\infty}^{\infty} f(x') g(x - x') dx',$$

and arises naturally in the analysis of imaging systems. If the imaging system is linear, $f(x)$ the true intensity distribution, and $g(x)$ the response of the system to a point source, the measured image is simply $f * g$.¹ That is, the convolution of the real sky with the point spread function (PSF) of the system is a blurring. The inversion of this blurring to recover the true image is called deconvolution, and is the subject of this dissertation. The deconvolution problem is naturally addressed in Fourier space, since the Fourier transform

¹The blurring is actually the correlation of the PSF with the true image, which is somewhat less tractable analytically than a convolution. When the difference matters, the problem is analysed as the convolution with the reflection of the PSF through the origin. In most interferometric applications, the PSF must be symmetric from fundamental considerations, and the difference between correlation and convolution disappears.

of the convolution of two functions is the product of their individual Fourier transforms: $\mathfrak{F}[f * g] = \mathfrak{F}[f] \mathfrak{F}[g]$, where \mathfrak{F} is the Fourier transform.² In fact, if the Fourier transform of the PSF is everywhere nonzero, and ignoring the question of measurement error, the direct solution to the deconvolution problem of determining f from measurements of $f * g$ and g is simply

$$f(x) = \mathfrak{F}^{-1} [\mathfrak{F}[f * g] / \mathfrak{F}[g]] \quad (1.1)$$

This dissertation primarily concerns interferometric imaging, where the Fourier transform of the image is measured with an interferometer. Each distinct baseline between two elements of the interferometer measures the image transform at a specific two-dimensional spatial frequency, (u, v) . The particular spatial frequency sampled is determined by the geometry of the array and source. In the commonly used technique of earth rotation synthesis, the relative geometry between array and source changes with time, leading to a more complete coverage of the Fourier plane. But even with this enhancement, the coverage of the Fourier plane is less than perfect. The sampling pattern $S(u, v)$, also called the u - v coverage, is used to quantify this and is usually a collection of Dirac delta functions located at the positions of the sampled spatial frequencies. The total information measured can be represented as SV , where $V(u, v)$ is the transform of the desired image. Inverse transforming the measured data to the image plane results in the basic convolution equation that must be inverted,

$$I^D = I * B, \quad (1.2)$$

where I^D is the measured dirty image, B is the PSF, and I is the true image that we wish to determine. I^D is the inverse or back transform of SV . B is the back transform of the sampling pattern alone, and is more commonly known as the synthesized dirty beam or point spread function. Since $S = \mathfrak{F}[B]$ is the analog of $\mathfrak{F}[g]$ in equation 1.1, the many zeros present in the typical sampling pattern show why the simple inverse filtering approach to deconvolution cannot be used. The unmeasured spatial frequencies are simply not present in the data, and cannot be recovered by any linear processing. The solution to this deconvolution equation involves estimating the image at the unmeasured frequencies, and makes implicit use of a priori information about the image. This a priori information can be that the sky is mostly empty, that the emission is sitting on an otherwise empty background, that the true image is constrained to be positive, or more sophisticated measures of image properties such as smoothness. Many good approaches exist for the solution of equation 1.2, most notably CLEAN & Maximum Entropy, but the synthesis imaging art is now well enough advanced that techniques which were good enough for an earlier generation of instruments are now found to be lacking in some cases.

²There are several sign and normalization conventions for the Fourier transform in common use. This dissertation uses Bracewell's "System 1," (Bracewell, 1986, p. 7), and the exact definitions are given in the list of symbols.

Some of the techniques developed here will also be applicable to filled aperture imaging, such as in conventional optical astronomy. There the convolution equation can arise as the convolution of the instrument response with the true sky or as a statistical degradation caused by the atmosphere. The physical reasons resulting in the convolution equation for filled aperture and interferometers can be the same, incomplete Fourier coverage, but they lead to qualitatively different kinds of “instrumental response”. The purpose of the deconvolution is typically different. In a filled aperture case, the PSF is usually well behaved with modest support, and one can often proceed without deconvolution. The usual reason for deconvolution is to increase the resolution of the image. By contrast, contemporary synthesis imaging instruments typically have Fourier plane coverages which result in sidelobes at the several to tens of percent level, extending out to infinity. One *must* deconvolve in order to uncover the structure beneath the sidelobes of the brighter sources in the field. The advantage of synthesis imaging is that the array geometry and consequently the PSF is known *a priori* to high precision, as opposed to the optical case where it must usually be measured or approximated. In addition, the Fourier components obtained in synthesis imaging can conveniently be reweighted to control attributes of the PSF, while at least some aspects of the PSF are frozen into the optical observation at the time the data is taken. Collectively, this has resulted in somewhat different attitudes to deconvolution in the filled aperture and synthesis imaging communities.

The dissertation title includes the adjective “High Fidelity.” Here this means both the traditional regime of high dynamic range imaging, where believable structure is sought around bright objects at peak to off-source RMS dynamic ranges of 10,000:1 or more, and also the *low* dynamic range regime of super resolution. If high fidelity means loosely “learning something about the source unrevealed by conventional image processing,” then super resolution is certainly high fidelity. It is simply that the conventional base image for comparison has little high resolution information and any believable high resolution information is by comparison “high fidelity.” Similarly, the definition of high fidelity is also somewhat stretched to encompass the new weighting techniques of Chapter 3. In some cases the resolution of the image is increased relative to conventional weightings, and that is certainly high fidelity as just discussed. In other cases the sensitivity limit due to thermal noise in the receivers is decreased, and this could again be called high fidelity when compared with the conventional higher noise images. The “moderately resolved” in the title refers fundamentally to the algorithms in the algebraic deconvolution chapter. Because my interest in the subject of high fidelity deconvolution was piqued with compact sources, the many simulations presented here address this size regime preferentially. For these purposes, a compact source is one with an area of a few hundred synthesized beam areas or less. The topic of visibility weighting, which comprises some half the length of this dissertation, is not limited to any particular source size — though again the particular simulations used to verify the deconvolution related properties of the weighting happen to use a compact source.

Beyond the obvious significant results of a new weighting and a new deconvolution algorithm, a third major conclusion of this work is simply that deconvolution errors can be significant, even when viewing quite simple emission, and that these errors can often masquerade as calibration errors. Imaging interferometers have become more capable since their introduction, and their calibration and use more sophisticated. Deconvolution algorithms which were adequate for earlier use are now found to be undesirable or unsuitable for some demanding modern projects. The magnitude of the deconvolution errors are larger than generally appreciated even on perfect data. The deconvolution errors grow worse if the algorithms are misused, generally meaning insufficient iteration. And the errors introduced by the interaction of calibration errors and deconvolution errors can be worse than the sum of the individual errors. This dissertation includes considerable effort trying to quantify these effects. The majority of the simulations address only the deconvolution aspects of the problem in a lower limit error philosophy, though some have been as end-to-end realistic as it is possible to do with contemporary software.

1.1 Overview of Dissertation Structure

The majority of the new techniques are introduced and developed in Chapters 3, 4 and 6, though some of the more minor “processing tricks” are mentioned in passing in the case studies of Chapters 7–9. Also, the mundane details of some of the weighting techniques are deferred to the appendices. An effort has been made to confine images from real data to the case study chapters, while the examples in the previous chapters are generally from synthetic data.

In Chapter 2, the basic equations of interferometric imaging are derived from first principles, following standard work. The discretization of the continuous equations is done more carefully than is usual, and the conventional delta function model of a pixel is generalized to include an arbitrary function as a pixel basis. Some limitations of the conventional even sized FFTs are discussed and a solution suggested. The chapter concludes with a discussion of deconvolution issues in general, temporarily ignoring the specifics of individual algorithms.

Chapter 3 is very long and covers the topic of visibility weighting. It is named after the primary result of robust weighting, but actually covers a number of weighting related topics, such as gridless weighting and beam forcing with a generalized Gaussian taper. This chapter is fairly complete and includes extensive case studies and advice on how to use existing tools effectively and combine them with the new techniques.

By contrast, the algebraic deconvolution Chapter 4 is a very promising beginning, but far from complete on the topics it addresses. The reintroduction of algebraic techniques to interferometric imaging has already yielded some spectacular examples compared to existing algorithms at the highest fidelity levels, but it is still very much an active research topic. This chapter addresses both Singular Value Decomposition deconvolution, which is

only adequate compared to existing methods, and Non-Negative Least Squares deconvolution, which is significantly better than traditional methods for certain regimes of image and source size. More work needs to be done in exploring the advantages and circumventing the limitations of these algorithms.

Chapter 5 is a large collection of simulation results, including an atlas of error morphologies for the major algorithms on simple sources, and quantitative error curves for many different algorithms on a wider variety of simple sources. The conclusions from this chapter are also considered a major result of this dissertation, as the magnitude of these effects had not previously been appreciated, nor the fact that deconvolution errors can often masquerade as calibration errors.

Chapter 6 is a miscellaneous collection of minor results and techniques that either did not fit anywhere else, or that would have been too serious a digression in the related discussion. This includes estimation of the thermal noise from the visibility data, a detailed discussion of how CLEAN places components around compact sources and the implications for model-fitting, and a discussion of how models from different deconvolution algorithms may be combined.

Chapter 7 begins the first of three case studies, presenting observational results from the VLA on the source 3C48. As this source was the one which inspired much of the work in this dissertation, a model of it has been used as a test source in several earlier chapters. Consequently this chapter is fairly short, with X band images showing that the weighting and deconvolution techniques do work as advertised on real data. A rather spectacular L band image shows both the ability of NNLS deconvolution to work on multiple isolated small sources in a large field and also the nonexistence of a potential extended jet at the 180,000:1 dynamic range level.

Chapter 8 concerns a super-resolved study of the source SN1987A with the Australia Telescope. The primary algorithm selected for this study was Maximum Entropy rather than NNLS, though both were used. The work concentrates on how one may best use existing tools for super-resolved imaging of simple sources, with emphasis on reliability and avoiding over interpretation. Astrophysically significant and believable structure is seen in SN1987A at a size scale of half the synthesized uniform beamwidth.

The last case study, Chapter 9 on the source DA193, was part of a test to explore the limits of the new Very Long Baseline Array. Earlier tests with CLEAN as the deconvolver had given hints that the compact core might be surrounded by a diffuse halo at the several thousand to one level. Deconvolutions using NNLS showed no such halo, and simulations of model observations showed NNLS capable of detecting such a halo were it present. The early NNLS results were later vindicated with noise limited observations at 112,000:1 which also show no halo. This work underscores the importance of NNLS in the hybrid mapping loop and the importance of gain error correlation structure in good simulations.

A short summary of the major results of this work is given in the conclusions Chapter 10. Appendices A–E all relate to the visibility weighting chapter, Appendix F reviews complex Fourier series for Chapter 6 and Appendix G concerns the software package used for most of the work in the dissertation, and the final Appendix H describes how to obtain several machine readable files containing more extensive simulation results.

Chapter 2

Representation of the Deconvolution Equation

This chapter is a mixture of review, extension, and clarification of the basic equations governing interferometric imaging. We begin with a standard derivation of the theoretical measured visibility, making manifest which fundamental assumptions are required for the particularly simple form of the imaging equations assumed in later chapters. Section 2.2 derives the form of the continuous dirty map, again following the standard derivation. In Section 2.3 we depart from the standard formalism in the discretization of the continuous equations. We introduce the concept of a continuous pixel model, a limiting case of which becomes the familiar CLEAN component. This formalism naturally leads to the conclusion that the component representation is a distinct space from either the image or visibility data. We demonstrate a problem with the standard representation and transformation of Hermetian data, and suggest a fix in the form of odd-sized FFTs. The chapter continues with a discussion of the general issues involved in deconvolution, and concludes with a list of the particular algorithms used in the dissertation along with a short description of each.

2.1 The Visibility Equations

In this review section we derive the expected interferometer response to the sky emission. We start from a very general formulation and introduce sufficient simplifying assumptions to derive the most common expressions for the observed visibilities. The review largely follows the excellent derivation by Clark (1989). Other modern derivations of similar and related topics can be found in Thompson (1989), Thompson, Moran, & Swenson (1986), and Christiansen & Högbom (1985).

The starting point of the formalism is a time-variable electric field of astrophysical origin at location \mathbf{R}_3 in three-dimensional space. This is $\mathbf{E}(\mathbf{R}_3, t)$. Maxwell's equations describe how an electromagnetic wave will propagate from location \mathbf{R}_3 to \mathbf{r} , the position of the later observation. The general time variability of the electric field vector introduces considerable complication into the later analysis, so it is most convenient to work in frequency space. Any finite interval of a real time varying function can be expressed in terms of a complex Fourier series. The function is the real part of the sum of complex coefficients times a simple time-varying exponential. We can expand each component of the electric

field in such a series, and write

$$\mathbf{E}(\mathbf{R}, t) = \operatorname{Re} \left(\sum_k \mathbf{E}_{\nu_k}(\mathbf{R}) e^{2\pi i \nu_k t} \right)$$

For our purposes Maxwell's equations are completely linear (Jackson, 1975, pp. 10–17), so we drop this summation and henceforth treat only a single *quasi-monochromatic component* of the electric field, \mathbf{E}_ν . Invoking that linearity, we superpose the contributions from all possible sources at the observed position \mathbf{r} .

$$\mathbf{E}_\nu = \iiint P_\nu(\mathbf{R}, \mathbf{r}) \mathbf{E}_\nu(\mathbf{R}) dx dy dz \quad (2.1)$$

The function $P_\nu(\mathbf{R}, \mathbf{r})$ is called the propagator, and describes how the electric field at \mathbf{R} affects the field at \mathbf{r} . In full generality, P must be a tensor function to properly account for the vector nature of \mathbf{E} . No polarization phenomena will be treated in this dissertation, so we may instead take both P and E_ν to be simple scalars.¹

The next simplifying assumption we make reflects the great distances to astronomical sources. We make no attempt to deduce the full three-dimensional spatial structure of the source, and instead consider the radiation on a given surface. We choose a great sphere of radius $|\mathbf{R}|$ centered on the origin, with the requirement that $|\mathbf{r}| \ll |\mathbf{R}|$.² With this choice, the electric field amplitude on that surface is simply related to the surface brightness. The geometry is given in Figure 2.1. Having assumed the celestial sphere, we next assume that space within it is empty. The propagator can be written down simply from Huygen's Principle, and equation 2.1 becomes

$$E_\nu(\mathbf{r}) = \int \mathcal{E}_\nu(\mathbf{R}) \frac{e^{2\pi i \nu |\mathbf{R} - \mathbf{r}|/c}}{|\mathbf{R} - \mathbf{r}|} dS, \quad (2.2)$$

where dS is the element of surface area on the celestial sphere.

Now consider the spatial correlation of the electromagnetic field. This correlation is defined at the points \mathbf{r}_1 and \mathbf{r}_2 by $V_\nu(\mathbf{r}_1, \mathbf{r}_2) \equiv \langle E_\nu(\mathbf{r}_1) E_\nu^*(\mathbf{r}_2) \rangle$. The raised asterisk is a complex conjugate, and the angle brackets indicate an expectation value over time. Substituting equation 2.2 for E_ν and rearranging the integrals over two separate surface elements, this becomes

$$V_\nu(\mathbf{r}_1, \mathbf{r}_2) = \left\langle \iint \mathcal{E}_\nu(\mathbf{R}_1) \mathcal{E}_\nu^*(\mathbf{R}_2) \frac{e^{2\pi i \nu |\mathbf{R}_1 - \mathbf{r}_1|/c}}{|\mathbf{R}_1 - \mathbf{r}_1|} \frac{e^{2\pi i \nu |\mathbf{R}_2 - \mathbf{r}_2|/c}}{|\mathbf{R}_2 - \mathbf{r}_2|} dS_1 dS_2 \right\rangle$$

¹Since propagators encountered in practice merely rotate the plane of polarization and preserve the total intensity, the scalar formalism above is exact for total intensity, even when observing intrinsically polarized sources.

²The actual requirement for synthesis imaging is that the angle formed by two telescopes and the object of interest be much less than a fringe spacing, or that $|\mathbf{R}| \gg |\mathbf{r}_1 - \mathbf{r}_2|/\lambda$.

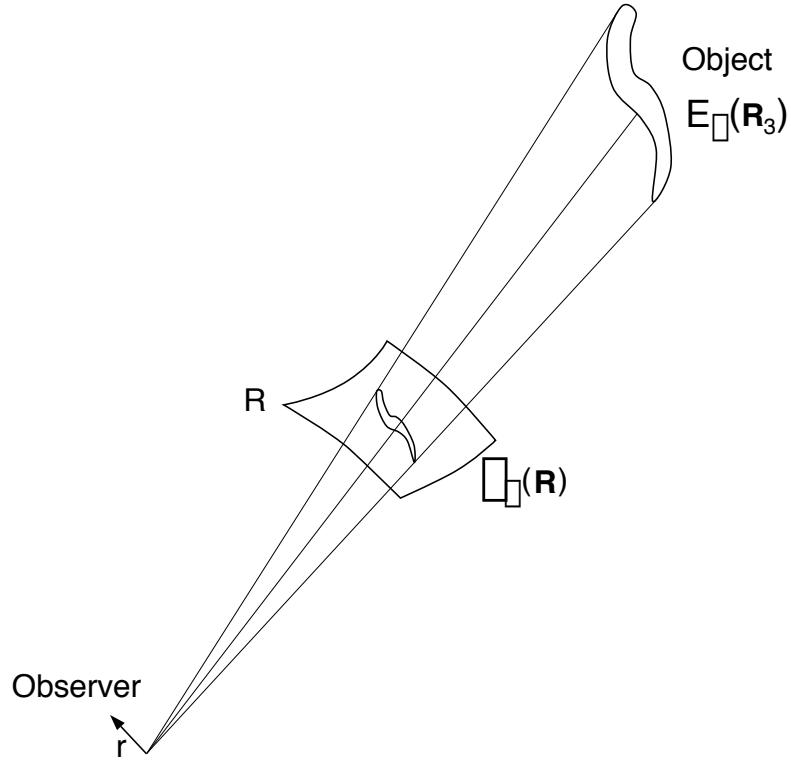


Figure 2.1: Radiation on the Celestial Sphere, adapted from Clark (1989). The only quantity we may measure with synthesis imaging is the equivalent radiation field on the surface of the celestial sphere, $\mathcal{E}(\mathbf{R})$.

The next simplifying assumption is that astronomical sources are not spatially coherent,³ that is

$$\langle \mathcal{E}_\nu(\mathbf{R}_1) \mathcal{E}_\nu(\mathbf{R}_2) \rangle = \langle |\mathcal{E}_\nu(\mathbf{R})|^2 \rangle \delta(\mathbf{R}_1 - \mathbf{R}_2)$$

For most objects, spatially incoherent emission is an excellent assumption. The notable exception occurs when a compact source is viewed through a scattering screen such as the interplanetary medium near the Sun. Special techniques must be used to image such sources, as detailed in Anantharamaiah et al. (1989). This will not be discussed further in this dissertation. Using the linearity of the expectation value, we exchange the expectation and integration to give

$$V_\nu(\mathbf{r}_1, \mathbf{r}_2) = \int \left\langle |\mathcal{E}_\nu(\mathbf{R})|^2 \right\rangle \frac{e^{2\pi i \nu |\mathbf{R} - \mathbf{r}_1|/c}}{|\mathbf{R} - \mathbf{r}_1|} \frac{e^{2\pi i \nu |\mathbf{R} - \mathbf{r}_2|/c}}{|\mathbf{R} - \mathbf{r}_2|} dS$$

³There is a significant detail glossed over here. It is not the radiation $\mathcal{E}(\mathbf{R})$ that is incoherent, but $\mathbf{E}(\mathbf{R})$. The free space propagation of the wavefront from the actual source to the celestial sphere does not change the coherence properties, however.

Let \mathbf{s} be the unit vector $\mathbf{R}/|\mathbf{R}|$. In a temporary coordinate system with the first axis pointed along \mathbf{s} , the coordinates of \mathbf{R} are $(|\mathbf{R}|, 0, 0)$, while the coordinates of \mathbf{r}_1 and \mathbf{r}_2 have the form $(\mathbf{s} \cdot \mathbf{r}_k, O(|\mathbf{r}|), O(|\mathbf{r}|))$. In this coordinate system, it is easily shown that

$$|\mathbf{R} - \mathbf{r}_1| - |\mathbf{R} - \mathbf{r}_2| = \mathbf{s} \cdot (\mathbf{r}_1 - \mathbf{r}_2) + O(|\mathbf{r}_1|/|\mathbf{R}|) + O(|\mathbf{r}_2|/|\mathbf{R}|)$$

Once again invoking the great distance to the celestial sphere to drop terms in $O(|\mathbf{r}|/|\mathbf{R}|)$, and defining the observed intensity as $I_\nu(\mathbf{s}) \equiv |\mathbf{R}|^2 \langle |\mathcal{E}_\nu(\mathbf{s})|^2 \rangle$, we have

$$V_\nu(\mathbf{r}_1 - \mathbf{r}_2) \approx \int I_\nu(\mathbf{s}) e^{-2\pi i \nu \mathbf{s} \cdot (\mathbf{r}_1 - \mathbf{r}_2)/c} d\Omega \quad (2.3)$$

Notice that equation 2.3 depends only on the baseline between \mathbf{r}_1 and \mathbf{r}_2 , and not on their absolute locations. (The primary reason for this is the assumption of spatial incoherence in the source.) Consequently we may learn all that can be observed about the correlation structure of the radiation field by holding one antenna fixed and moving the other about. There is no need to measure all possible pairs of points. Sampling at a translation of a baseline will yield the same information about the sky as sampling at the original baseline.⁴ The function $V_\nu(\mathbf{r}_1 - \mathbf{r}_2)$ is the *spatial coherence function* of the field $E_\nu(\mathbf{r})$.

Equation 2.3 is quite fundamental, and could in principle be used to compare a model of the sky emission with the measured visibilities. For the more usual purposes of synthesis imaging, we must make a final simplification based on observation geometry. There are two different observation geometries where the inversion of equation 2.3 becomes particularly tractable, and both are embodied by instruments used in this dissertation.

Suppose first that all of the measured vector spacings lie in a two-dimensional plane. That is, for some particular coordinate system all baselines can be represented in the form $\mathbf{r}_1 - \mathbf{r}_2 = \lambda(u, v, 0)$, where $\lambda = c/\nu$. Since \mathbf{s} is a unit vector, it can be written as $(\ell, m, \sqrt{1 - \ell^2 - m^2})$, where ℓ and m are direction cosines relative to the u and v axes. In this coordinate system, the differential of solid angle, $d\Omega$, is $d\ell dm / \sqrt{1 - \ell^2 - m^2}$. See Thompson, Moran & Swenson (p. 80, 1986) for the details of the differential's derivation. Equation 2.3 now becomes

$$V_\nu(u, v, 0) = \iint I_\nu(\ell, m) \frac{e^{-2\pi i (u\ell + vm)}}{\sqrt{1 - \ell^2 - m^2}} d\ell dm \quad (2.4)$$

This is a Fourier Transform equation for the modified intensity $I_\nu(\ell, m)/\sqrt{1 - \ell^2 - m^2}$. Most connected element interferometer arrays on the surface of the earth lie in a two-dimensional plane or very close to one. Hence all baselines between antennas also lie in a plane. For these arrays, any instantaneous measurements can be processed with this formalism, and no

⁴This property is exploited in devices with “redundant” baselines. By enforcing the constraint that the measured coherence functions on such baselines are equal, one can learn about the calibration errors to which the device is subject.

other assumptions about geometry need be taken into account. For earth rotation synthesis instruments, however, the baselines will rotate about an axis parallel to that of the earth. (Visibilities are invariant to baseline translation.) The only arrays which will remain planar after earth rotation synthesis are east-west arrays, which are popular for this and other reasons. In Chapter 8 we present data and images from the Australia Telescope compact array, which is just such an instrument.

A similarly useful form of the visibility equations results when all sources contributing to the visibility are in a small region of the sky. This form is used when processing data from intrinsically three-dimensional arrays like the VLBA or two-dimensional arrays like the VLA used in earth rotation synthesis mode. Data from both these arrays are used extensively in later chapters. A convenient way to quantify the “small region” concept is to define $\mathbf{s} = \mathbf{s}_0 + \boldsymbol{\sigma}$ and ignore any terms of order $|\boldsymbol{\sigma}|^2$. An immediate consequence of the fact that \mathbf{s} and \mathbf{s}_0 are unit vectors is that

$$1 = \mathbf{s} \cdot \mathbf{s} = \mathbf{s}_0 \cdot \mathbf{s}_0 = \mathbf{s}_0 \cdot \mathbf{s}_0 + 2\mathbf{s}_0 \cdot \boldsymbol{\sigma} + \boldsymbol{\sigma} \cdot \boldsymbol{\sigma} \approx 1 + 2\mathbf{s}_0 \cdot \boldsymbol{\sigma}, \quad (2.5)$$

so that \mathbf{s}_0 and $\boldsymbol{\sigma}$ are orthogonal. If we choose the w axis to lie along \mathbf{s}_0 , we have $\mathbf{r}_1 - \mathbf{r}_2 = \lambda(u, v, w)$, $\mathbf{s}_0 = (0, 0, 1)$ and $\boldsymbol{\sigma} \approx (\ell, m, 0)$. Again ignoring terms of order $|\boldsymbol{\sigma}|^2$, we drop the $\sqrt{1 - \ell^2 - m^2}$ term in the differential of area⁵ and equation 2.3 becomes

$$V'_\nu(u, v, w) = e^{-2\pi i w} \iint I_\nu(\ell, m) e^{-2\pi i (u\ell + vm)} d\ell dm \quad (2.6)$$

It is conventional to absorb the leading exponential term into the left hand side, by considering the modified quantity $V_\nu(u, v, w) \equiv e^{2\pi i w} V'_\nu(u, v, w)$. The resulting equation is independent of w .

$$V_\nu(u, v) = \iint I_\nu(\ell, m) e^{-2\pi i (u\ell + vm)} d\ell dm \quad (2.7)$$

Finally, we have to account for the directivity of the individual antenna elements. The individual elements of a radio interferometer do not measure the electric field at a single point, but rather have an angular sensitivity to the direction of the radiation’s propagation. Thus in equation 2.1 the integral should be weighted by the *antenna directivity*, $\mathcal{A}_\nu(\mathbf{s})$, which affects all subsequent equations to this point. The antenna directivity can also be called the *primary beam* or *normalized reception pattern*. In particular, equation 2.7 becomes

$$V_\nu(u, v) = \iint \mathcal{A}_\nu(\ell, m) I_\nu(\ell, m) e^{-2\pi i (u\ell + vm)} d\ell dm \quad (2.8)$$

⁵Alternatively, this factor may be grouped with the antenna directivity, $\mathcal{A}_\nu(\ell, m)$, in equation 2.8. This is not terribly important, since the dominant source of approximation error comes from the treatment of the exponential.

The quantity $V_\nu(u, v)$ is called the *complex visibility* relative to the *phase tracking center*.⁶ If the two antenna elements are dissimilar, as is frequently found in Very Long Baseline Interferometry, the appropriate directivity correction is geometric mean of the two individual antenna patterns, $\sqrt{A_{1\nu}(\ell, m) A_{2\nu}(\ell, m)}$. VLBI is also the case where the sources imaged are often sufficiently small that the effects of the individual primary beams can be ignored all together.

Equation 2.7 is the simplest form of the visibility equation, and will be used for all subsequent derivations. The alternative geometrical assumption and the correction for the antenna elements simply modify the form of the effective sky brightness. All corrections of this form should be deferred until the final stage of image processing, at which point they are trivially accounted for. It should be noted that while the antenna directivity appears to be a nuisance factor, it is this directivity that allows the small region of sky approximation to be widely used.

In principle, the instrumental response could be made to approach that of an ideal quasi-monochromatic component by observing with a negligibly small bandwidth. In practice, observations are often made with a fractional bandwidth of a few percent, and the measured visibility is reasonably well described by the monochromatic visibility at the center of the band. The visibility amplitude is modulated by the transform of the instrumental bandpass. The effect in the image plane is to smear sources radially towards the delay center, with the magnitude of the smear varying linearly with radius. This smear can be largely removed in software, but such fixes are rarely necessary. A more direct cure is to observe in spectral mode, where the effective bandwidth is divided by the number of spectral channels. In such a mode, bandwidth smearing is usually quite negligible. The effect of finite averaging time per visibility measurement also produces a smear whose magnitude varies linearly with radius from the delay center, but the smear direction is tangential. Time averaging is considered in the context of visibility weighting; otherwise, neither of these two effects are considered further in this dissertation.

⁶There are a number of fiducial points on the sky used in interferometry. The *pointing center* of the array elements is fixed at the time of the observation. The *delay tracking center* is fixed at correlation time, and is that point on the sky where the effective path length after compensation is the same for all antennas. Bandwidth smearing effects are centered on the delay center. The *phase tracking center* is defined as that point for which a point source has zero expected visibility phase on all baselines. This may be shifted after the fact by merely adjusting the phase of the visibility measurements. The phase center determines the center of the synthesized images. The *tangent point* is determined by the vector s_0 in equation 2.5, and concerns the mapping from the curved celestial sphere to Cartesian coordinates. Geometric distortions will be least near the tangent point. The tangent point may also be shifted after the fact but requires recalculation of the u, v , and w coordinates as well as the phase terms. In simple observations, all four of these positions are the same point. However in a mosaiced observation one may use multiple pointing centers to cover a large object. In a VLBI experiment, bandwidth effects may limit the field of view to a small region about the delay center, and the data may be correlated more than once with different delay centers. The phase center is shifted routinely to move interesting positions to the center of the images. If the shift is large enough that geometric distortion is important, the tangent point may be moved as well.

Summarizing, the simplifications necessary to go from the most general formulation of the electric field to the simple Fourier transform formalism are

- *Analysis in pseudo-monochromatic components* Essentially no approximation necessary
- *Celestial sphere* Cannot get depth information about source
- *Polarization phenomena ignored* Not needed here
- *Space is empty* No propagation effects
- *No spatial coherence* Very rarely violated
- *Measurements confined to a plane* or *All sources in small region*
These assumptions can be relaxed in software

2.2 The Continuous Dirty Map

Equation 2.7 is obviously a Fourier transform relationship. Were $V(u, v)$ known completely, we could simply use the direct inversion for $I(\ell, m)$, namely

$$I(\ell, m) = \iint V(u, v) e^{2\pi i(\ell u + m v)} du dv \quad (2.9)$$

However, a major difficulty is that the visibility function is only sampled incompletely on the u - v plane. We introduce a sampling function, $S_V(u, v)$, which is zero where no data has been taken. A convenient factorization of this function which reflects the discrete nature of the sampling process is

$$S_V(u, v) \equiv \sum_{k=1}^{N_V} w_k \delta(u - u_k) \delta(v - v_k) \quad (2.10)$$

where the u_k and v_k are the coordinates of the sampled data. Note that w_k is a weight here, and not the third Fourier coordinate. The notation is unfortunate, but widely used. w_k as a Fourier coordinate will not subsequently be used in this dissertation. The choice of w_k is completely arbitrary, and selection of these weights allows the user to adjust the balance between angular resolution, sidelobes of the beam and sensitivity. This topic will be covered extensively in Chapter 3.

We now have sufficient information to calculate a quantity we will call the continuous dirty map. (The terms map and image are used synonymously in this dissertation, as are dirty beam and point spread function.)

$$I^D(\ell, m) = \iint V(u, v) S(u, v) e^{2\pi i(\ell u + m v)} du dv \quad (2.11)$$

The convolution theorem for Fourier transforms immediately leads to the conclusion that

$$I^D = I * B \quad (2.12)$$

where

$$B(\ell, m) = \iint S_V(u, v) e^{2\pi i(u\ell + vm)} du dv \quad (2.13)$$

B is called the synthesized dirty beam, and in general will contain many sidelobes and unpleasant artifacts of the sampling. It is precisely this convolution of the dirty beam with the original sky brightness that we wish to invert with the deconvolution algorithms. Since the sky brightness is known to be real, the visibility data is constrained to be Hermitian. Thus without loss of generality, we may take $S_V(u, v)$ to be symmetric about reflection in the origin. This property is then shared by the synthesized beam, which will turn out to be quite important later. Note also that even before this equation is discretized, there is no possibility that the deconvolution is unique. Any function $\hat{V}(u, v)$ which is equal to $V(u, v)$ on the support of $S(u, v)$ will have a Fourier transform $\hat{I}(\ell, m)$ which satisfies the convolution equation. It is the introduction of additional information about the source by the deconvolution algorithms which allows us to select between the alternatives.

2.3 Discretization of the Continuous Equations

Even given the discrete nature of the visibility sampling process, the resulting convolution equation is continuous. To obtain a set of equations that can be manipulated by a computer, we need to discretize the continuous equation. Here we do the discretization in some detail. The standard formalism is generalized to allow for a more general ‘pixel model’ than the usual scaled delta function. For simplicity of notation, we return to the one-dimensional case. This is not limiting, in that the argument below can be directly generalized to the two-dimensional case, much as described in Appendix A of Andrews & Hunt (1977). We begin with equation 2.12, $I^D = I * B$. To invert this convolution, we shall find some deconvolution model function, C , such that $I^D = C * B$. Under some circumstances, C may have a direct relationship to I , but for the moment this equation should be taken as the sole definition of C . In particular, we should not blindly apply physical arguments about the positivity or support of I to C . The name C is to suggest a Component model, and later on the scalar coefficients of this model will become the familiar CLEAN Components. Remember, however, that $C(\ell)$ is a continuous function.

To discretize this equation, we define an image sampling function in analogy to the visibility sampling function defined earlier. Since this function will be used only to force agreement at the sampled points, there is no need for normalized weights.

$$S_I \equiv \sum_{i=1}^{N_\ell} \delta(\ell - \ell_{I,i}) \quad (2.14)$$

Most often, we discretize to a regular grid, in which case S_I becomes an ordinary shah function, $\text{III}(\ell/\Delta\ell)$. See Bracewell (p. 77, 1977) for properties of III. This is solely for computational convenience, however. The only fundamental limit is set by the criterion that the sampled visibility must not be aliased. That is, we require that

$$[V(u)S_V(u)] * \mathfrak{F}[S_I(\ell)] = V(u)S_V(u) \quad (2.15)$$

on the support of $S_V(u)$. For the finite, regular sampling case, this will be satisfied automatically if the maximum value of u on the grid, \hat{u}_{max} , is greater than the maximum value of u present in the data, u_{max} . (\hat{u}_{max} is set by the frequency of sampling in the image plane, $\hat{u}_{max} = 1/2\Delta\ell$.) Incidentally, it is this equation which allows us to ignore the high spatial frequencies which may be present on the sky. The aliasing constraint applies to $V(u)S_V(u)$ not merely $V(u)$. By definition, we have that

$$\begin{aligned} I^D S_I &= (C * B) S_I \\ I^D(\ell_{Ii}) &= (C * B)(\ell_{Ii}) \quad (i = 1, \dots, N_\ell) \\ &= \int_{-\infty}^{\infty} C(\ell') B(\ell_{Ii} - \ell') d\ell' \end{aligned} \quad (2.16)$$

In general, this is an integral equation, and is difficult to solve. To progress further, we must assume a form for $C(\ell')$. It is this arbitrary assumption about the form of C that severs the link between C and I . We may choose a form for C whose transform is complete in the visibility plane, (any possible measured visibility function may be represented in this form), but there is no guarantee that I or any given function of I may be representable in the form of C . This is one basis for the statement, *CLEAN components model the visibility data, not the image data*. Under some circumstances, it may develop that C becomes a known function of I , but this is not guaranteed. A second justification for this statement will be given later in this chapter, and yet a third in Chapter 6.

Since the Fourier transform is linear, it is convenient to form a model from a linear combination of components. Each component can be transformed independently into either domain. Indeed, this property is crucial to the operation of CLEAN and most deconvolution algorithms. The most general model consistent with our obtaining a discrete convolution equation is a sum of shifted and scaled replications of some basic pixel model. While a more complicated model could be used, the algorithm becomes a general model fitting procedure, and this formalism is no longer appropriate. We thus assume the following form for C

$$\begin{aligned} C(\ell) &\equiv \sum_{j=1}^{N_C} C_j P(\ell - \ell_{Cj}) \\ &= \sum_{j=1}^{N_C} C_j (P(\ell) * \delta(\ell - \ell_{Cj})) \end{aligned} \quad (2.17)$$

The term $C_j P(\ell - \ell_{Cj})$ is a model component (or generalized CLEAN component) of magnitude C_j , centered at the location ℓ_{Cj} . The function $P(\ell)$ is the pixel model. If

we take $P(\ell) = \delta(\ell)$, we recover the usual formalism. In addition, the ℓ_{Cj} are usually constrained to be integral multiples of some $\Delta\ell$. If the ℓ_{Cj} are determined in advance, (such as the above constraint to a regular grid), then the convolution process will reduce to a single matrix multiplication, and the determination of the C_j requires the conceptual (pseudo) inversion of this matrix. In some deconvolution algorithms like CLEAN, the determination of the C_j are done iteratively, and the location of the nonzero C_j are determined from the possible set $\{\ell_{Cj}\}$ “on the fly.” The CLEAN algorithm can be extended⁷ to determine the *unconstrained* ℓ_{Cj} on the fly, which leads to the “fractional cell CLEAN.” This modification is potentially fairly important, but the algorithm no longer fits into the matrix formalism and is quite difficult to analyze. Assuming that the ℓ_{Cj} have been predetermined in some manner, with N potential positions, then equation 2.16 becomes

$$\begin{aligned}
 & \text{(for } i = 1, \dots, N_\ell) \\
 (C * B)(\ell_I i) &= \left(\left(\sum_{j=1}^{N_C} C_j (P(\ell) * \delta(\ell - \ell_{Cj})) \right) * B \right) (\ell_I i) \\
 &= \int_{-\infty}^{\infty} \sum_{j=1}^{N_C} C_j \int_{-\infty}^{\infty} P(\ell'') \delta(\ell' - \ell_{Cj} - \ell'') d\ell'' B(\ell_I i - \ell') d\ell' \quad (2.18) \\
 &= \sum_{j=1}^{N_C} C_j \int_{-\infty}^{\infty} P(\ell'') B(\ell_I i - \ell_{Cj} - \ell'') d\ell''
 \end{aligned}$$

If we write

$$B(\ell_I i - \ell_{Cj} - \ell'') = B(\ell' - \ell'') * \delta(\ell' - \ell_I i + \ell_{Cj})$$

then we have that

$$\begin{aligned}
 (C * B)(\ell_I i) &= \sum_{j=1}^{N_C} C_j \int_{-\infty}^{\infty} P(\ell'') \int_{-\infty}^{\infty} B(\ell' - \ell'') \delta(\ell_I i - \ell_{Cj} - \ell') d\ell' d\ell'' \\
 &= \sum_{j=1}^{N_C} C_j (P(\ell) * B(\ell))(\ell_I i - \ell_{Cj}). \quad (2.19)
 \end{aligned}$$

We can identify this last equation as a matrix equation of the form $\mathbf{Ax} = \mathbf{b}$.

⁷“Extended” is actually the wrong word here. Implementations of exactly this local fitting procedure have been around nearly as long as the original algorithm. But in modern usage it has fallen somewhat out of favor and there is no literature on its analysis — the regular spacing CLEAN is difficult enough to analyse!

$$\underbrace{\left[\begin{array}{c} P * B \\ \vdots \\ P * B \end{array} \right]}_{N_\ell \text{ columns}} \left\{ \begin{array}{c} C_j \\ \vdots \\ C_j \end{array} \right\}_{N_\ell \text{ rows}} = \left[\begin{array}{c} C * B \\ \vdots \\ C * B \end{array} \right]_{N_C \text{ rows}} \quad (2.20)$$

P is not necessarily symmetric, although B must be so. Thus in general, $P * B$ will not possess any particular symmetries. More usually, P , B , and $P * B$ will all be symmetric. We will write $P * B$ as B' and refer to it as the modified beam matrix. In the standard case of δ function components, $P = \delta * B = B$.

2.4 Convolutional Gridding and Aliasing

The formalism is now complete to the extent that the equations could be solved with a sufficiently powerful computer. We would need to know an analytic form for the inverse Fourier Transform of P , and apply it to all the weights w_k . We would then have to find the forward transform of both the modified weights and the unmodified visibility samples $V(u_k)$, the beam being evaluated at all points $\ell_I i - \ell_C j$, and the dirty map at all points $\ell_I i$. Remembering that $(C * B)(\ell_I i) = (I * B)(\ell_I i)$ by definition, we insert these values into equation 2.19, and solve the resulting set of (probably singular) linear equations for the C_j .

The problem with this direct approach is that it is a formidable computational challenge for contemporary problems. It is not unthinkable large — we will use exactly this algebraic approach in Chapter 4 and the computation of Fourier transforms by direct summation over visibilities is practical for some problems, if too slow to be used in an iterative manner. We would very much like to use a Fast Fourier Transform, which offers savings in computation times of order $N/\log_2 N$, where N is the number of pixels in the transform. The use of an FFT for the computation requires that the data be evenly spaced, and that the number of elements be highly composite. The evenly spaced criterion leads us to choose $\ell_I i = i\Delta\ell$ and $\ell_C j = j\Delta\ell$.⁸ Immediately we get that $\ell_I i - \ell_C j = (i - j)\Delta\ell$, and the matrix $P * B$ becomes Toeplitz in form. Symmetries in the FFT further force the form of the matrix to a symmetric circulant matrix. This introduces additional approximations which can either be tolerated or eliminated by embedding $P * B$ into a larger matrix of twice the size. But most important of all is that the data in *both* domains must be regularly spaced. We had the freedom to choose where we enforced agreement between the dirty component model and the dirty sky. We had the freedom to choose where we allowed

⁸Obviously this can lead to some creative indexing. $i = 0 \Rightarrow \ell_I i = 0$ is the center of the map.

model components to be placed. But we have (in most cases) no control at all over the detailed placement of the visibility sampling.

The visibilities are usually forced to a grid by evaluating at the grid points a convolution of the visibilities with some function of modest support. To first order, this convolution is simply a multiplication by the transform of the gridding convolution function in the image plane. As such, it is easily divided out again. The problem enters because the convolved function is resampled in the u - v plane and it is the transform of the *resampled* function which is corrected for the original convolution. The aliasing introduced by the resampling cannot be divided out. Modern mapping programs use a convolution function whose transform has a fairly sharp cutoff at the edge of the image plane. This provides at least two orders of magnitude isolation at the edge of the map between aliased power and its image in the final corrected map. The map center typically has four orders of magnitude isolation. The details of the convolutional gridding and the consequent aliasing can be found in Schramek & Schwab (1989) and references therein. We will not be particularly concerned with aliasing in this dissertation as most of the simulations are run with models discretized in the image plane where gridding is not an issue. In the remainder of the image processing, we are usually concerned with compact sources far from the edge of the images where aliasing is the worst. There have been no significant observed discrepancies which could be attributed to convolutional gridding or other aliasing effects. The other property that the use of FFTs forces upon us, that the vector length be highly composite, is worth a bit more comment.

2.5 Limitations of the even-sized FFT

Since Fourier transforms are central to most of our image processing algorithms, it is surprising that there are still some fundamental facts about the discrete Fourier transform of Hermitian data that are unappreciated in the interferometry community.

With the common choice of u - v origin on a pixel center and the standard power-of-2 restriction on FFT size, the discrete sky and visibility planes are not exact Fourier transform pairs!

The problem is that if the u - v origin is at the center of a pixel, as is normally desired, the imaginary part of that pixel is constrained to be zero since the image plane data is real. The Real \leftrightarrow Imaginary FFT routines that we use⁹ are strictly reciprocal, so the number of degrees of freedom in the data set is reduced by one and must appear elsewhere. The last degree of freedom appears as the real part of the highest spatial frequency. That is, if the real data spans $(-N/2 + 1, -N/2, \dots, N/2)\Delta\ell$, the Fourier data will span $(0, 1, \dots, N/2)\Delta u$, but the imaginary part of the $u = 0$ and the $u = N/2\Delta u$ samples will

⁹SDE uses the routines `rfftf` and `rfftbt` from (Swarztrauber, 1985).

be forced to zero by the mechanics of the transform. One can see that this is plausible by remembering that the real property of Fourier transformed Hermitian data comes from the symmetry property $V(-u) = V^*(u)$. In the discrete transform, every element of the vector is conceptually paired with a conjugate pixel on the other side of the origin, out to the same length as the real vector. The conjugate pixel is redundant, of course, and not actually stored. But pixel corresponding to $u = N/2 \Delta\ell$ has no conjugate element, as that would result in a vector longer than the original real vector. If the output is to be real, $V(N/2 \Delta\ell)$ must be forced real. When going in the direction Real \rightarrow Complex, this last element will simply appear with zero imaginary part, or will be packed in with the first pixel and the application programmer sets the imaginary part to zero when using it. But in going the Complex \rightarrow Real direction, the best that can be hoped is that the imaginary part of that pixel will be ignored or that the mechanics of the packing implicitly assume it to be so. In some cases, however, the FFT routine will actually get the wrong answer when the imaginary part of the last pixel is nonzero. And in no case will one be able to fill the full complex vector with nonzero values, transform to the real domain and back and preserve the last pixel. In this sense, the transform is nonreciprocal, though of course from degrees of freedom arguments, no transform could be so on vectors of even size.

We normally do not see this effect, since in the cases where we start in the Fourier domain, the sampling pattern of the interferometer forces the last row and column of the gridded data to be zero or very nearly so. In cases where we are transforming data with nonzero values out to the edge of the Fourier plane, the data has probably been generated in the image plane as model components, and the requirement that the last row and column be real is enforced implicitly in the Real \rightarrow Complex transform. The only time this would be noticed is if one started in the Fourier domain with a model, transformed to the real domain and back and compared the results to the original. I know of no algorithms currently in routine use where this operation is important. In situations where it does matter, the effect can be avoided by the use of an odd-sized FFT. The $2N + 1$ real values exactly map to the $2(N + 1) - 1$ values in the complex vector with the imaginary part of the origin pixel held at zero. In the next section, it's shown that one need not pay an excessive computational price for the use of an odd-sized vector.

2.5.1 Non-Power-of-2 FFT Performance Analysis

Conventional wisdom usually holds that Fast Fourier Transforms are most efficient when the size of the data vector is a power of 2. This is indeed true, but such knowledge has often been unfortunately extrapolated to mean “greatly more efficient” than transforms on other sizes. Surprisingly, this is not necessarily true. Figure 2.2 shows the measured execution times vs data size for one particularly good scalar FFT package that is widely used and freely available, (Swarztrauber, 1985). The upper panel shows that there are clearly many bad choices of data size. As would be expected, the uppermost trace is the line of primes, and succeeding lower traces are generated by numbers of the form $2p$, $3p$

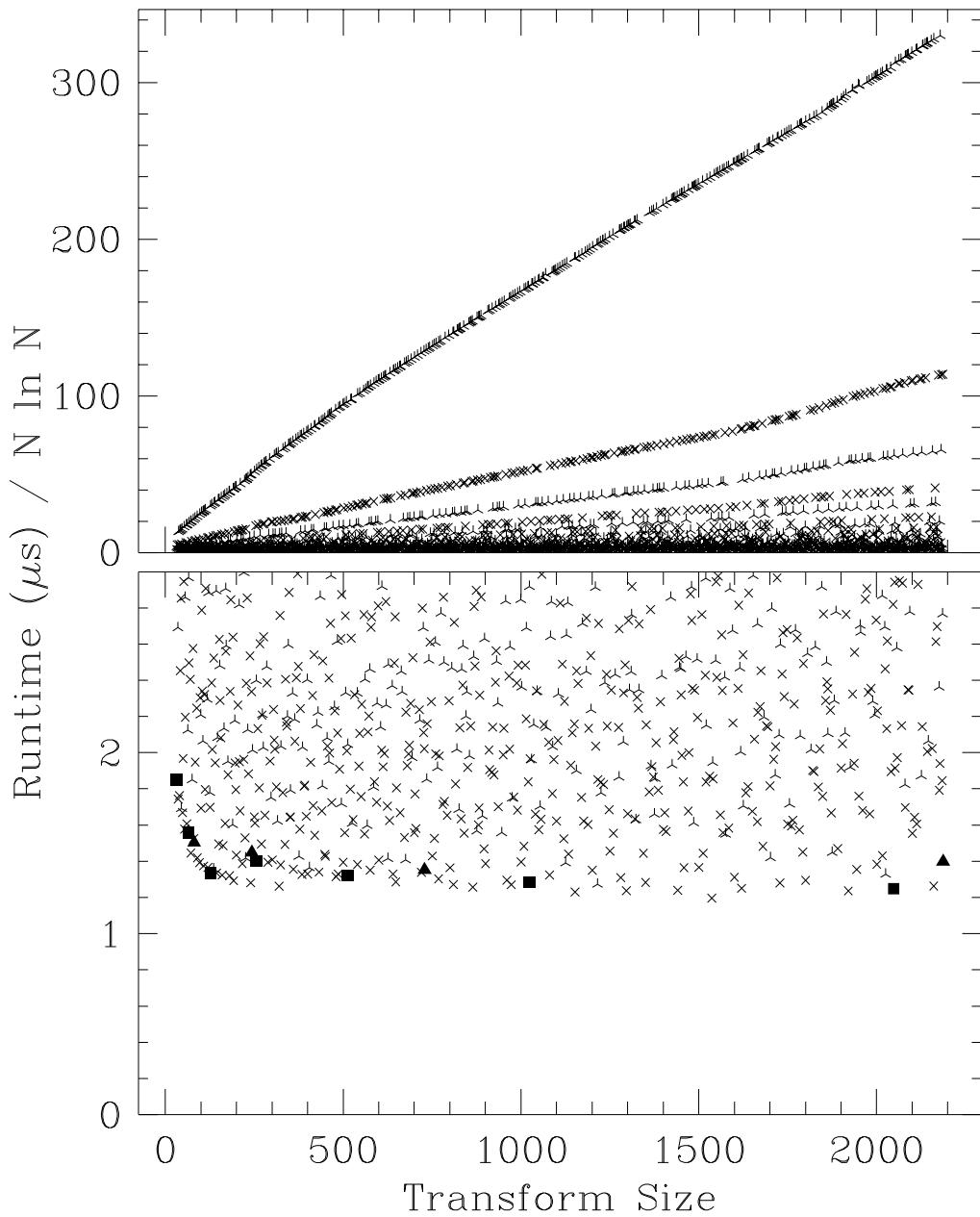


Figure 2.2: Normalized FFT runtimes vs. data size. The $O(N \log N)$ asymptotic behavior of the algorithm has been divided out of the measured runtime. Average runtimes were measured on an unloaded Sparc IPX running the subroutines `CFFTI`, `CFFT`, and `CFTTB` from (Swarztrauber, 1985). Even sizes are represented with x symbols, and odd sizes with three pronged stars. The filled symbols are pure powers of 2 and 3.

and so forth. However, if we examine the lower panel, we see that there are many sizes that are nearly as efficient as 2^N . The runtime of this algorithm on vectors of size 2^N was comparable or better than that of several common algorithms restricted to this form. The lower envelope of Figure 2.2 was drawn by hand, and this was used to determine the efficiency loss for a given data size in the obvious manner.

From this, we estimate that if one picks a random odd number within the examined range of 32–2187, one has only a 1.8% chance of selecting a size with an efficiency loss less than 25%. Even if one were to deem an efficiency loss of 100% acceptable, doubling the runtime, one has only a 12.5% chance of selecting such an odd number by chance. However, the distribution of highly efficient odd numbers is such that one can typically select an efficient data size no more than 5–10% larger than the actual data vector. If the size of the data vector is fixed in advance, the need for padding to the next higher power of 2, (with its attendant potential for aliasing), is reduced or eliminated. Similarly the non-Hermetian effects described in this section are avoided. Since the one-dimensional FFT runtime scales like $N \ln N$, saving an average of 50% in the necessary vector length more than outweighs the typical efficiency loss of 15% seen for odd sizes.

Two-dimensional FFTs can be viewed as iterated one-dimensional FFTs — they are often calculated by taking the FFT of each column and then subsequently taking the FFT of each row. Native two-dimensional FFTs exist, and are usually more efficient by a small constant factor in execution time than the iterated one-dimensional FFT. The scaling of $N \log N$ along each axis still holds and the execution time of either approach for a square $N \times N$ image scales as $N^2 \log N$. I do not know if a good two-dimensional FFT algorithm exists for an arbitrary length along each axis, (**SDE** uses iterated one-dimensional FFTs), so the execution time argument here becomes somewhat weaker compared to two-dimensional power-of-two FFTs if a two-dimensional arbitrary-size routine is not available. But notice that for large images or even large image cubes, the savings in disk space alone may be sufficient to justify iterated one-dimensional arbitrary-size FFTs. Table 2.5.1 lists all highly efficient odd sizes in the range 33–2187, as measured on a scalar workstation. These numbers are not necessarily correct for more exotic machine architectures, but still indicate that the performance issues should be examined. Table 2.5.1 lists all efficient sizes in that range, both even and odd, but does not show factorization or loss.

One may object that image sizes are often picked for convenience, and that for many purposes the exact field of view of an image is not terribly important. In such situations, researchers habitually choose image sizes of the form 2^N and padding losses do not apply. However, in this case it is just as easy to pick a convenient odd image size as 2^N . For example with images sizes of 405^2 or 625^2 instead of 512^2 , the efficiency loss is negligible and one gains the benefit of an unambiguous image center and freedom from Hermetian concerns. Consider that nearly any image processing package will place the reference pixel of a $2k + 1$ pixel axis at the $k + 1$ th pixel, but for a $2k$ pixel axis the choice is not clear. Different packages use either the k th or the $k + 1$ th pixel, and **AIPS** even uses

a different convention on each axis. This makes transportation of images between packages more difficult for no particularly good reason.

There is a practical danger in the use of arbitrary-N FFTs in contemporary packages. Power of 2 FFTs have dominated the field for so long that library subroutines often assume without documentation that image sizes are even or 2^N . Assumptions about the location of the phase center of the transform can be subtle and deeply imbedded in the code.¹⁰ Consequently one may have less confidence in FFT based results from these packages when using arbitrary-N FFTs. The images in this dissertation have generally been made with sizes of 2^N for this reason, since there seemed no problem in these cases with the Hermetian representation, and images sizes were otherwise arbitrary. But transition uncertainties should not blind us to the fact that arbitrary-N FFTs are clearly superior for some purposes, and insistence on 2^N image sizes is archaic. Even if older software is not converted, we must strive to avoid introducing this restriction into future packages.

There is no reason to restrict FFT data sizes to powers of 2, and occasionally good reasons to avoid it.

¹⁰Even worse, in packages such as SDE which transform into the complex half plane, the size of the image plane axis conjugate to the short Fourier plane axis may not be stored explicitly. If image sizes (prior to entering the FFT) are constrained to be even, this may be recovered by calculation. When this constraint is relaxed, there is a fundamental loss of information and the backtransformed image size is uncertain by one. Retrofitting such older software can be difficult.

N	<i>factorization</i>	<i>Loss</i>	N	<i>factorization</i>	<i>Loss</i>
45	3 3 5	0.2	625	5 5 5 5	10.6
81	3 3 3 3	3.7	675	3 3 3 5 5	10.9
135	3 3 3 5	2.1	729	3 3 3 3 3 3	6.9
189	3 3 3 7	23.5	1125	3 3 5 5 5	13.6
225	3 3 5 5	10.5	1215	3 3 3 3 3 5	4.5
243	3 3 3 3 3	13.4	1701	3 3 3 3 3 7	17.1
375	3 5 5 5	11.9	1875	3 5 5 5 5	18.7
405	3 3 3 3 5	6.7	2025	3 3 3 3 5 5	7.9
525	3 5 5 7	23.6	2187	3 3 3 3 3 3 3	11.9
567	3 3 3 3 7	23.6			

Table 2.1: Odd FFT sizes between 33 and 2187 with efficiency losses less than 25% of optimal.

32	80	150	225	360	480	600	750	972	1215	1536	1920
36	81	160	240	375	486	625	756	1000	1250	1600	1944
40	90	162	243	378	500	640	768	1008	1280	1620	2000
45	96	168	256	384	504	648	800	1024	1296	1680	2016
48	100	180	270	400	512	672	810	1080	1344	1701	2025
50	108	189	288	405	525	675	840	1120	1350	1728	2048
54	120	192	300	420	540	700	864	1125	1440	1792	2160
60	128	200	320	432	560	704	896	1134	1458	1800	2187
64	135	216	324	448	567	720	900	1152	1500	1875	
72	144	224	336	450	576	729	960	1200	1512	1890	

Table 2.2: All FFT sizes between 32 and 2187 with efficiency loss less than 25% of optimal.

2.6 Deconvolution

We conclude the chapter with a discussion of the basic issues involved in deconvolution algorithms, followed by a list of the particular algorithms used in this dissertation and a short description of each one.

All deconvolution algorithms accumulate some model of the sky¹¹ either iteratively or via a direct solution. The model is convolved with the point spread function and differenced with the dirty image, yielding the residuals. The sky model may possess nonzero spatial frequencies out to the maximum gridded spatial frequency determined by the pixel spacing on the sky, or the spectral power may roll off at the higher spatial frequencies due to a smoothness constraint imposed by the particular algorithm. In either event the model generally possesses power at frequencies not measured, both interior to and exterior to the maximum measured spatial frequency. The spatial frequencies estimated beyond the sampling envelope are somewhat suspect and instead of trying to recover *the* model which exactly reproduces the data at *all* possible spatial frequencies, we generally agree to seek *a* model which yields a good reconstruction of the sky at some finite resolution. Consequently, the model produced by the deconvolution algorithm is usually smoothed down by convolution in the image plane with a Gaussian “restoring beam” fitted to the main lobe of the point spread function. This is a multiplication of the model by a Gaussian in the Fourier domain. In practical terms, the highest measured spatial frequency is usually down-weighted to of order 10% the unsmoothed value for a uniformly weighted PSF, and to less than a few percent if the restoring beam is fitted to the naturally weighted PSF. In order to give a reasonable indication of the thermal noise present in the data and account for the unmodelled residual flux so far as possible, the final restored image is usually formed from the smoothed model plus the residuals. The practice of adding the residuals back to the restored image is now nearly universal in the radio community, but was not so in the past. The optical community seems not yet to have come to a consensus, with some practitioners publishing the raw model from the deconvolution algorithms and others following the modern radio practice.

There are a few subtleties in this general process. First, the truth image sought by the algorithm is the actual sky smoothed down to a finite resolution. But more specifically, it is the smoothed true sky sampled at the image grid points, and not the sampled sky which is then smoothed. Clearly, something like this must be the case, or a point source between pixels would yield zero. In many of the simulations in this dissertation, the distinction is dodged entirely by discretizing the model in the image plane and convolving with the discretized point spread function yielding a dirty image entirely free of representation

¹¹Once again, I prefer to consider the component model as a representation of the visibility data, not the sky. None the less, the term “sky model” is too vivid a description to discard entirely. It simply must be remembered when discussing aliasing or detailed representation questions that the term is a convenient abbreviation, and not the literal truth.

effects. In others, the analytic model used to generate the visibilities was smoothed analytically and the resulting model discretized to provide the comparison image. Comparing the deconvolved images to a sampled and smoothed model made a considerable difference and was clearly the wrong approach.

There is no really fundamental reason why the restoring beam should be fitted to the main lobe of the PSF. If all the flux in the source has been transferred to the model, then all the restoring beam does is express how much confidence we have in the high spatial frequency estimates. In general, using a smaller restoring beam will allow more of the extrapolated high spatial frequencies into the solution and the deconvolution errors will become worse. (This statement is not new, but one of the results of this dissertation is a striking demonstration of how dramatic an effect it is.) There is not even any particular reason that the restoring beam should be a Gaussian. One obvious choice might be the function which is unity inside the sampling envelope and zero outside it. In such a case, the restoring beam would allow estimates of the visibilities at spatial frequencies where they can be interpolated, but would not allow any extrapolation at all. Obviously, in such a case the restoring beam would look roughly as bad as the dirty beam itself, so such a literal approach would not be useful. But the principle remains that we do not solve for the sky so much as for a known function of the sky. One tries to select a function which minimizes the error in reconstruction yet which allows the extraction of as much useful information from the image as possible — but the choice is still an arbitrary one.

The one property that a fitted beam does have as a restoring beam is that it preserves the flux scale so much as is possible on the residuals, in the event that not all the flux is transferred to the model. The only “flux scale” that the residuals or the dirty image have is that a point source of magnitude S will have a peak response of S on the dirty image. No similar statement can be made at all about extended sources, which is one very good argument for deconvolution in the first place! By contrast, the units of the deconvolution model are usually Janskys per pixel.¹² The same point source will now have magnitude S in the component model as well, but it will only occupy one pixel rather than a collection of them as in the dirty image. After smoothing the model with the restoring beam, the scale of the smoothed components is adjusted so that the point source maintains a peak response of S in the restored image, the units of which are now given as Janskys per restoring beam. The result of this procedure is that to the extent that the dirty beam “looks like” its fitted Gaussian, a point source which is deconvolved will resemble a point source which was not and which enters the restored image via the residuals. One often estimates the total flux density in a source by integrating up the response in the restored image and dividing by the volume of the restoring beam. If one chooses a tight integration

¹²1 Jy = 10^{-26} W/Hz/m², and is the standard unit of radio astronomy. Note that it is really a unit of flux density rather than flux, but speaking of “flux density per synthesized beam” quickly becomes tiresome, and one often speaks of “flux scales” rather than “flux density scales” as shorthand.

window around a compact source and uses a fitted restoring beam, this procedure will not be *too* far wrong if the flux has entered the restored image via the residuals. In that sense, a fitted restoring beam preserves the flux scale.

There is also the question of how finely the dirty image and dirty beam must be sampled. People tend to measure the degree of oversampling in an image in “points per beam.” It is a useful concept and a reasonable rule of thumb — especially if the restoring beam is a fitted one. But implicit in this practice is that there exists a unique beam for a given sampling and weighting type. In signal processing, the canonical “two points per cycle” rule is derived from the requirement that the highest frequency in the data be representable in the transform of the discrete time series, namely that the $f_{max} < 1/2\Delta t$. In the case of interferometric data, we know the maximum spatial frequency present in the data. The criterion for critical sampling is then inverted, yielding our $\Delta\ell < 1/2u_{max}$. For a given data set, this number is fixed, and does not vary with weighting or tapering. The fitted beam size does vary, however, and the two criteria are consequently not equivalent. Table 3.1 in Section 3.4 will later show some explicit examples of how widely the beam size can vary just by changing the image and cell size in conjunction with uniform weighting.

There are good reasons for wanting to sample the image more finely than the simple requirement of critical sampling. Critical sampling guarantees only that information is not lost going from one plane to the other, not that it is in a particularly useful form. In a critically sampled time series, for instance, one can recover the original continuous band limited signal by sinc function interpolation. Each interpolated point becomes a complicated alternating series with many terms which do not die off very quickly with distance from the interpolation point. It is not obvious looking at two adjacent pixels in a critically sampled series what an intermediate value should be, since it depends so heavily on nonlocal information. A standard way to do this interpolation, in fact, is to transform into the frequency domain, pad with zeros, and back transform into the time domain. This, of course, is exactly what we do when we set the image cell size smaller than the critical sampling value; the measured visibilities are padded with zeros in the frequency domain. As the series or image is sampled more finely, the interpolation series becomes simpler and eventually approaches linear interpolation. Visually, the images become much easier to interpret. A more compelling reason to oversample an image comes from the fact that the nonlinear deconvolution algorithms will create unmeasured spatial frequencies in the data, extrapolating out past u_{max} and v_{max} . Loosely speaking, the algorithms need somewhere to put the power that they generate, or the model will become corrupted by edge effects. In general, high oversampling is a good idea for high fidelity imaging, and for such projects I usually operate in the 6 points per “typical” uniform beam regime, or roughly one third of the critically sampled limit. For super resolution projects I may sample more finely still.

Finally, as a generalization to the standard techniques, note how any deconvolution algorithm can be generalized to use an arbitrary pixel model. Equation 2.19 showed

how the fundamental deconvolution equation is modified when using a general pixel function, namely $I^D = I * (B * P)$, which reduces to the usual form when P is chosen to be a delta function. In terms of how we use it in a deconvolution algorithm, it means we must convolve the dirty beam (but not the dirty image) with the desired pixel model prior to doing the deconvolution. The deconvolution model and residuals are found in the standard way. The values in the component model are now simply the coefficients C_j in equation 2.17, so the model coefficients are convolved with the pixel function P prior to being smoothed with the restoring beam. The residuals are then added back as usual. Notice that for a generalized model $C' * P$ to represent the same visibility data as the conventional model C , the high spatial frequency model coefficients must be amplified by the inverse of the pixel function transform. The added high frequency spectral power present during the conventional portion of the deconvolution process may be a destabilizing factor for some algorithms.

2.7 Deconvolution Algorithms

The following algorithms have been used to deconvolve data in this dissertation. With the sole exception of the Cotton-Schwab CLEAN, all of these algorithms operate in the sampled image plane or its discrete transform. That is, the visibility data is used only to form the dirty image and point spread function. In some cases the SDE task `uvmap` was used for this purpose. In others, an existing PSF was convolved with a sampled model to form a dirty image free of aliasing and representation effects, using the tasks `imgmodel` and `imgconv`.

2.7.1 CLEAN

The CLEAN algorithm was devised by J. Högbom in 1974, (Högbom, 1974), and represents the sky by a number of point sources on an otherwise empty background. The algorithm proceeds iteratively, and chooses the peak of the residuals at each iteration as the point of highest correlation between the PSF and the residuals.¹³ At the location of the peak residual, a fraction of the value there is accepted as a CLEAN component. Specifically, the dirty beam B , normalized to a unit maximum, is multiplied by the peak residual and a damping constant, $\gamma \leq 1$. γ is termed the loop gain. This scaled beam is shifted to the location of the maximum residual, subtracted from residuals, and the corresponding component added to a list. In this way, flux is incrementally transferred from the residuals to an equivalent CLEAN component list. The process is iterated until the maximum residual is less than some user defined threshold, or to some maximum number of iterations. Support information can be incorporated by limiting the region where the algorithm is allowed to

¹³This is only strictly true for the uniformly weighted PSF, but in practice is approximately so for all point spread functions encountered in radio interferometry.

search for the maximum residual. As described before, the component model is smoothed with a restoring beam and summed with the final residuals to form the restored image. For such a simple algorithm, the results have been quite spectacular. In many cases, the quality of the solution produced holds up well in comparison to those of algorithms developed two decades later.

Beyond the simple Högbom algorithm, there are two major variations which perform essentially the same algorithm, but do so in a more efficient way. The first is the Clark CLEAN, (Clark, 1980), which introduced the concept of major and minor cycles. It uses the insight that since the repeated shift, scale and add cycle of the Högbom algorithm is essentially convolution of the CLEAN components with the PSF, the process may be improved with the use of FFTs. The minor cycle proceeds largely as in the Högbom algorithm, except that the only residuals considered are those with an intensity greater than α times the peak residual, where α is the fractional height of the PSF's worst exterior sidelobe. The processing is done in a list based format, and the interaction between pixels further limited by considering only a region of the beam near the center, called the beam patch. The result of all this is an approximation to the CLEAN components that would have been determined by the Högbom algorithm, but one that can be computed comparatively quickly. When the residuals have been CLEANed deeply enough that the approximation starts to break down, the algorithm enters a major cycle and recomputes the residuals of the entire CLEAN component model accumulated to that point, via a single FFT pair. The model is transformed to the Fourier domain, multiplied by the weighted sampling function which is the Fourier transform of the PSF, back transformed to the image plane, and differenced with the original dirty image. In this way, errors introduced by the approximations in one minor cycle can be corrected in subsequent cycles.

As the goal of the Clark CLEAN is to duplicate the Högbom CLEAN, at cost in additional approximation, the Clark CLEAN has not been used significantly in this dissertation. In its defense, the cross comparisons done between Clark and Högbom have shown little to no difference in most cases. But given that computational resources existed to compute the original Högbom formulation, I felt there was no need to introduce an additional source of approximation into these experiments. Note that the Clark CLEAN, and any iterative algorithm that periodically returns to the original data or dirty image, is more resistant to accumulated rounding errors than is the Högbom CLEAN. This has not been found a significant problem for fewer than 10^6 components or so. Experiments at the 10^7 component level have created artifacts due to roundoff error, and use of an algorithm which utilizes major/minor cycles would be well advised at such levels. This does not affect any result presented here. The term CLEAN, without qualification, will mean the Högbom CLEAN in this dissertation. The implementation used was the `Hogbom` algorithm within the `SDE` task, `clean`.

The second variant form of the CLEAN algorithm in common use is the Cotton-Schwab algorithm, (Schwab, 1984), which is embodied in the `AIPS` task `MX`. As with the

Clark CLEAN, this is a major-minor cycle iteration, with the difference that during a major cycle the model components are subtracted not from the original dirty image, but from the original visibility data. This is a considerable improvement in that aliasing due to the convolutional gridding is nearly eliminated. It also allows multiple fields to be imaged simultaneously. Each field undergoes the minor cycle independently, and its distant sidelobes are removed from the other fields during the major cycle. Finally, when the residuals are computed, the geometric approximations described in Section 2.1 can be reduced by proper consideration of the w term for each subfield. For many real-world imaging problems, Cotton-Schwab CLEAN is the algorithm of choice. The particular sources imaged in this dissertation did not require use of a Cotton-Schwab style algorithm, as single fields were sufficient to cover all emission, and for compact sources near the field center aliasing is not a major concern. Only a few tests were performed with MX directly. However, the error modes of the Högbom CLEAN described here will affect any more sophisticated algorithms which uses it as a minor cycle.

The best reference on the practical use of CLEAN in deconvolving radio images is Cornwell & Braun (1989). It also features a short review of CLEAN's theoretical properties, not given here. The seminal paper on the theoretical analysis of CLEAN is Schwarz (1978), which establishes several necessary criteria for the algorithm's convergence. This is followed by the more general review paper on CLEAN, Schwarz (1979). For a somewhat pessimistic extension of Schwarz's work, covering the initial phases of the iteration as well as the asymptotic limit, see Tan (1986). Marsh & Richardson (1987) features an analysis yielding the objective function minimized by the CLEAN algorithm in the particular case of imaging an isolated point source, but a result for the more general case is not known. A proper noise analysis for the general case is also lacking for CLEAN. Much has certainly been done in the theoretical understanding of the algorithm, but the situation must still be called unsatisfying. This dissertation is largely a pragmatic one and will certainly add to the "lore" of the algorithm, but beyond a few very minor analytic results presents no new analysis on the fundamentals of the algorithm. For our purposes here, the most important properties of the algorithm to note are that the CLEAN components may be positive or negative, and that there is nothing in the algorithm which enforces smoothness of the model.

2.7.2 SDI CLEAN

The Steer-Dewdney-Ito CLEAN, (Steer et al., 1984), is also a variety of CLEAN, but one which diverges more radically from the original Högbom form than the algorithms just described. The algorithm is one of several which attempts to avoid the production of stripes in the CLEAN map, as described by Schwarz (1984). The solution adopted in SDI CLEAN is simply to shave off all residuals greater than some trim threshold times the maximum residual and accept them as components. If the original source is smooth, the residual map will largely be so and thus the model components as well. The components selected in the trim stage must be scaled in some manner to account for the different effects

of the dirty beam on resolved and unresolved sources. One possibility is to convolve the potential components by the dirty beam and scale the components such that the peak of the dirty components is equal to the peak dirty residual. The rescaled components are then multiplied by a conventional loop gain and the new residual image formed just as in a major cycle of the Clark CLEAN. The implementation used, algorithm **SDI** within the **SDE** program **clean**, actually scales the model components to minimize the RMS mismatch between the dirty components and the selected residuals. The algorithm described by Steer et. al. scales to the peak residual and also includes a provision for processing point sources separately from the main emission. Neither of these differences is believed to affect the results presented here.

2.7.3 Multi Resolution CLEAN

Since CLEAN is known to work better on compact emission, a natural extension is one where a low resolution version of the image is processed first to obtain the more extended structure, and a higher resolution image produced by working on the difference image. A bonus is that fewer components are needed to represent the large scale structure when working at lower resolution, and very often the signal-to-noise characteristics are more favorable as well. The same general approach can also be repeated on a number of different size scales, rather than a single sum and difference map. Such an algorithm has been devised by Wakker & Schwarz (1988). The dual resolution version of their algorithm has been implemented in **SDE** as the task **mrc**. This formalism was not appropriate for the single source resolution tests, though it was attempted on the 3C48 model source. Disappointingly, it fared no better than standard CLEAN at the highest fidelity levels. This algorithm will not be used otherwise in this dissertation, though it is certainly promising for many problems and deserves much wider exposure than it has seen.

2.7.4 Maximum Entropy

The second major algorithm which is a standard staple of radio interferometric deconvolution is Maximum Entropy. In contrast to CLEAN which is procedurally defined, MEM is a maximization over a single criterion. The image is selected which fits the visibility data to within the noise, and which also maximizes the “entropy.” The precise definition of and theoretical justification for the form of the entropy is a matter of considerable dispute. Here we merely note that most of the proposed forms give similar results for the problems we are interested in. In particular, the **SDE** task **vm** maximizes the entropy form

$$\mathcal{H} = - \sum_k I_k \ln \frac{I_k}{M_k},$$

where M_k is a default model image allowing for the incorporation of a priori information about the source. Support information may be incorporated by limiting the region where

the sum is evaluated, and allowing only those pixels to vary. If the default image is constant, as it was in all images presented in this dissertation, the entropy functions as a smoothness constraint. The form of the entropy itself forces the image values to be nonnegative. All maximum entropy methods which purport to optimize the same criterion should yield the same result. The problem of finding the global maximum is difficult enough, however, that a number of different algorithms have been proposed for its solution, both public and proprietary. This dissertation exclusively uses the Cornwell and Evans algorithm, as coded in `vm`. There are some suggestions that the extremely demanding applications here may be violating some of the approximations in the algorithm, leading to exceedingly long converge times for the highest precision images. The Maximum Entropy deconvolutions given here could be redone with other MEM implementations, to see if the convergence speed is increased. It is believed that the final results given here would be essentially unchanged.

Also in contrast to CLEAN, the literature on Maximum Entropy is voluminous. A good practical reference on the use of MEM in deconvolving radio images is once again Cornwell & Braun (1989). The excellent review article Narayan & Nityananda (1986) is highly recommended, and includes a fairly detailed yet pragmatic overview of the theoretical properties of the MEM solution. Additional papers presenting opinions on the proper form of the entropy and its possible physical interpretation include Frieden (1972), Wernecke & D'Addario (1976), Gull & Daniel (1978), Jaynes (1978), Narayan & Nityananda (1984), and Cornwell & Evans (1985). Algorithms for the solution of the Maximum Entropy problem are given in Wernecke & D'Addario (1976), Cornwell & Evans (1985) and Skilling & Bryan (1984).

2.7.5 Non-Negative Least Squares

NNLS deconvolution is described in detail in Chapter 4 of this work. The basic idea behind the deconvolver is extremely simple. The solution minimizes the squared norm of the residual vector, subject to the constraint that the model components are positive. Unlike other approaches which include a least squares criterion and positivity, the equations are not solved iteratively. The full matrix connecting a window on the component image with a window on the dirty map is formed, and subsequently solved with conventional linear algebraic numerical methods, namely the subroutine `NNLS` from Lawson & Hanson (1974). The support information is encoded directly in the structure of the linear equations representing the beam, and is necessary both to improve the quality of the reconstruction and to reduce the computation to a manageable size. The solution is guaranteed to satisfy the condition that the partial derivatives of the squared norm against positive model pixels are zero, and against zero pixels point in the direction of negative flux. That is, the solution is a local minimum of the objective in the allowed pixels, and the gradient is orthogonal to the constraints. Given the highly singular nature of the linear systems generated by typical interferometric deconvolution equations, there is no guarantee that there will be a

unique solution to the problem, even with the positivity and support constraints. Still, the solutions produced by the particular algorithm used seem to be remarkably stable for compact emission. While quite computationally and memory intensive, this algorithm produces the best deconvolutions known on compact objects and noiseless data. The desirability of including positivity in a direct solution of the linear convolution equations has been known for quite some time, (Andrews and Hunt, 1977, p. 188), but until this dissertation these methods appear not to have been applied to radio astronomical deconvolution.

2.7.6 Maximum Emptiness

Marsh and Richardson (1987) showed that for the case of an isolated point source, the CLEAN algorithm approximately minimizes the sum of the pixel values in the component model subject to the constraint that CLEAN components are positive. That is, it minimizes the L1 norm of the positivity constrained components and yields a minimum flux or maximum emptiness solution. By comparison to empirical results, clearly this analysis is only an approximation, even in simple cases. It did inspire the creation of an experimental deconvolver by Tim Cornwell. With the Cornwell-Evans Maximum Entropy deconvolver conveniently at hand, it was quite simple to modify the optimization criterion to a softened version of the L1 norm. The result is a deconvolver which shares some of the properties of both CLEAN and MEM. In actual practice it does not seem to be a significant improvement over either, but it is interesting and useful in deciding which features of the model might be driven by the mechanics of the optimization and which might derive from the actual properties of the optimization criterion itself. There is no formal literature reference on this algorithm.

2.7.7 Gerchberg-Saxon-Papoulis

The GSP algorithm is conceptually very attractive in that it is simple and one can see immediately what information is going into the solution. The algorithm merely alternates between the image and Fourier planes, enforcing support and positivity and biasing the solution towards agreement with the data. If one is mistrustful of structure produced from more complicated criteria, it is comforting when the GSP iteration returns a similar result. As implemented in the SDE task `gsp`, the iteration is simply: (1) Set all model pixels outside the support window to 0. (2) Add a small fraction of the residuals to the model image, forcing the model towards agreement with the data. (3) Multiply all pixels less than zero by $\alpha \leq 0$. In the usual case that α is 0, this is a simple clip and enforcement of positivity. (4) Re-form the residuals from the modified model and test for convergence. If not done, return to (1). The fundamental references for this algorithm are Gerchberg & Saxton (1972) and Papoulis (1975). It is slower than many algorithms, and tends to require good support information for convergence, but is still surprisingly effective.

2.7.8 Richardson-Lucy

The Richardson-Lucy algorithm is highly favored in the optical community for a number of reasons. The algorithm converges to the maximum likelihood solution for Poisson statistics in the data, which is appropriate for optical data with noise from counting statistics. The RL solution does not change the total flux in the model, which is good from the point of view of someone possessing a lossless optical system such as a defocused filled aperture telescope, but unfortunate for interferometry where one rarely has a reliable total flux. In the case of the SDE implementation, `lucy`, the total flux is left arbitrary during the iteration, and after the solution has converged the flux scale is found by minimizing the RMS mismatch between the dirty component model and the original dirty image. The RL method interprets the PSF as the probability distribution for scattering a photon from the true location to the observed one. The probability of seeing a photon at a given pixel is written as the sum over all pixels of the probability of seeing a photon at an arbitrary pixel, times the conditional probability of scattering it from that pixel to the one of interest. Bayes' theorem is used to write the sum as an equivalent one involving the pixel of interest as an explicit leading factor, which leads to a simple iteration. Alternatively, one can start from the maximum likelihood estimator for Poisson statistics, and directly derive the RL iteration. Both derivations are appealingly simple from the optical perspective, but the method is unfortunate in that it requires the PSF to be positive. The draconian approach to adapting this deconvolution algorithm to radio astronomy is simply to clip both the PSF and the dirty image to positive values before entering the RL iterations. In spite of the fact that the clipped dirty image, dirty beam and true object distribution no longer obey the convolution equation, this procedure does often yield reasonable results. Clearly, the Richardson-Lucy algorithm is not an ideal match to the needs of radio interferometry. But again, it is comforting when algorithms based on very different criteria yield similar results. The Richardson-Lucy algorithm is used for exactly this reason in Chapter 8, where the super-resolved deconvolution problem is sufficiently delicate that many different algorithms were tried in an attempt to decide if certain super-resolved structure was believable. Beyond the primary references Richardson (1972) and Lucy (1974), a good overview of how the algorithm is used at the Space Telescope Science Institute is given in White (1993).

2.7.9 Singular Value Decomposition

Pseudo-inverse filtering via a Singular Value Decomposition of the PSF is a standard technique, and is described in Andrews & Hunt (p. 164, 1977). When applied to the interferometric deconvolution problem with no support constraints, the PSF is diagonalized by the discrete Fourier Transform matrix, and the singular values of the PSF are just the gridded visibility weights. The pseudo-inverse solution to the convolution equation is simply the uniformly weighted principal solution, and thus not particularly useful. In Chapter 4, it is shown that when the SVD is augmented with support information about the source, the

shape of the singular value spectrum changes, and the cutoff in singular value used in determining an approximate pseudo-inverse serves as a regularization parameter. As with the NNLS algorithm, the support information is encoded in the structure of the windowed beam matrix. Determining an appropriate cutoff is rather arbitrary, the computational demands considerable, and the quality of the solution returned only comparable with other methods. By itself this is not a particularly good deconvolution method. But it is demonstrative of the role support information plays in regularizing the solution, it provides another algorithmic cross comparison method for use on difficult projects, and in the future it might prove useful combined with other algebraic methods. This method is used only in Chapter 4 and also in the supernova project of Chapter 8.

With the general issues and preliminaries out of the way, we can now proceed to a much more specific and detailed treatment of visibility weighting.

Chapter 3

Robust Weighting

3.1 Introduction

A new form of visibility weighting is presented that varies smoothly from natural to uniform weighting as a function of a single real parameter, the robustness. The name of the parameter refers to a striping instability that can be generated by uniform weighting. Natural weighting is immune to this problem and hence most ‘robust’. While originally derived as a cure for the striping problem, robust weighting has several additional desirable properties.

Intermediate values of the robustness can produce images with substantially improved thermal noise characteristics at very little cost in resolution. The robust weighting gains its thermal performance by varying the effective weighting as a function of local u - v weight density. In regions where the weight density is low, the effective weighting is natural. In regions where the weight density is high, the effective weighting is uniform. The combined effect is to smooth the large variations in effective weight produced by uniform weighting, resulting in a more efficient use of the data and a lower thermal noise. Robust weighting has an elegant derivation as the solution to a minimization problem. The standard weightings are derived here from minimization principles, and the robust weighting then derived as a natural extension of this.

Robust weighting can reduce the RMS thermal noise of a typical full track VLA uniformly weighted map by 25%, while increasing the width of the fitted beam by only 3%. In the particularly well suited cases of a badly unbalanced VLBI array, the weighting can produce RMS improvements of a factor of 2 at a 16% loss in resolution. Larger values of the robustness can produce beams that have very nearly the same point source sensitivity as the naturally weighted beam, but with enhanced resolution. In some cases, this intermediate beam also has dramatically improved sidelobe characteristics. For the VLA in particular, this intermediate resolution beam has high sensitivity, low sidelobes, and little of the extended ‘shelf’ usually found in VLA naturally weighted beams. It is an excellent choice for general purpose mapping of extended objects. The value of the robustness which produces this beam can either be selected by eye, or as the result of a procedure which minimizes the mismatch between the beam and the fitted Gaussian. Robust weighting does not help as much with snapshot u - v coverage, as there is relatively little difference between uniform and natural weightings. Improvements made with other techniques are significant, however.

An alternative method of stabilizing the uniformly weighted beam against striping is also given in this chapter. It removes the weighting irregularities imposed by the standard gridded reweighting, but is still vulnerable to irregularities caused by pathological u - v sampling. It offers some of the same thermal noise improvements as does modest robustness. Its primary virtue beyond robustness is simplicity. The astronomer need only specify gridless weighting, and need not worry about choosing an appropriate robustness. This method can be combined with robustness, but the additional increase in sensitivity from combining the methods is small. Gridless weighting works particularly well on sparse u - v coverages.

Finally, the interaction between these new approaches and Gaussian (inverse) tapering is examined. Tapering is an occasionally overlooked method of beam control, and it works well with the methods presented here.

3.2 Thermal noise in the synthesized map

Consider a standard derivation of the noise in an ideal synthesized map, such as that found in Walker (1989) or Thompson, Moran & Swenson (1986). We consider only the thermal noise, and write the dirty image as the Fourier transform of the weighted, sampled, noise-corrupted visibilities:

$$\begin{aligned} I^D(\ell, m) &= C \mathfrak{F}[V^W] = C \mathfrak{F}[VV] \\ &= C \sum_{k=1}^{2L} T_k D_k w_k V_k e^{2\pi i(u_k \ell + v_k m)} \end{aligned} \quad (3.1)$$

where

$$W(u, v) = \sum_{k=1}^{2L} T_k D_k w_k \delta(u - u_k, v - v_k)$$

$V_k = \mathcal{V}_k + \varepsilon_k$ is the measured complex visibility at (u_k, v_k) , with \mathcal{V}_k the true visibility and ε_k the complex noise. C is a normalization constant, T_k is a tapering function, w_k reflects the signal-to-noise ratio of the data point, and D_k is the density weight introduced to control various properties of the image. The factors T_k , D_k , and w_k are in principle arbitrary, but they are factored out separately for convenience in later manipulation. We will call $W_k \equiv T_k D_k w_k$ the total weight. The sum is written with $2L$ terms to indicate the presence of the Hermitian conjugate visibilities. For every visibility $V_k(u_k, v_k)$ in the summation there is a conjugate term $V_k^*(-u_k, -v_k) = V_k^*(-u_k, -v_k)$. All weights are the same for both conjugations of the visibility, in order to assure that the sky image is real.

The weighting situation is somewhat complicated in contemporary imaging packages, in that traditionally but a single weight is associated with each visibility. In different contexts, this is interpreted as different combinations of T_k , D_k , and w_k , though most often it is simply w_k . Worse, there is still another weight that is associated with the quality of

the calibration. Some programs have been known to modify the visibility weights to reflect the scatter of the visibilities in the antenna based calibration solution. There is simply no way for all programs to treat the weights consistently in all contexts, as several independent quantities are being shoehorned together into a single storage quantity. For instance, most imaging programs calculate the T_k and D_k on the fly from some small set of analytic parameters entered by the user. However, if a program were to calculate some specialized version of the D_k to pass onto a later generalized imaging program there would be no place to put them without sacrificing w_k . The situation is somewhat more cheerful for **AIPS++**, in that a minimum of w_k will be factored out into a separate storage area, though the precise model has not yet been specified. For the moment, we must simply accept that existing solutions will be cumbersome, and keep firmly in mind which quantities are stored where at what points in the analysis.

In these derivations, we make two major simplifying assumptions. The first is to restrict ourselves to the noise dominated regime, where calibration errors are negligible. More specifically, we assume that the real and imaginary part of each visibility measurement is an unbiased estimator of the source visibility, and that both parts are contaminated with independent realizations of the same Gaussian random variable. Such thermal noise is well behaved and easy to measure in practice. For a given instrument and observing conditions, the thermal variance in the correlated signal is proportional to $1/\Delta t$, where Δt is the integration time. For experiments involving different instruments, as in a mixed VLBI array, we assume that the proportionality constants can be measured such that the relative variances in the visibilities are accessible to the investigator.¹

It can be shown (see Appendix A) that when combining different measurements of the same quantity to form a weighted mean, the minimum variance in the mean is obtained by weighting inversely with the variance of each sample. Consequently in equation 3.1 we call w_k the signal-to-noise-ratio weight, and assume without loss of generality that it is proportional to the inverse variance of the thermal noise in each visibility. In the case of the VLA, this is nearly² true by default. In the VLBI case, it is usually true after the data has been calibrated. In such cases, we say that the weights $w_k \propto 1/\sigma_k^2$ are statistical. Occasionally we need to convert explicitly from weights to variance and back. The constant ΔS_{uw} is the RMS noise on a single correlator channel corresponding to a visibility of unit weight. Thus $w_k = \Delta S_{uw}^2 / \sigma_k^2$; ΔS_{uw}^2 is the proportionality constant between w_k and the inverse variance $1/\sigma_k^2$. $\sigma_k = \langle \text{Re}^2 \varepsilon \rangle^{1/2} = \langle \text{Im}^2 \varepsilon \rangle^{1/2}$ is exactly the quantity ΔS calculated in

¹High frequency instruments like the Sub-Millimeter Array can also produce nonuniform thermal variance due to the atmosphere. As the instrument tracks the source across the sky, the atmospheric contribution to the system temperature varies in an analytically estimable way which should be reflected in the snr weights.

²Ed Fomalont reports an improvement of 5% in naturally weighted RMS after adjusting an X band data set to properly reflect the antenna to antenna variations in receiver temperature, rather than using the normal assumption that all telescope are identical.

the standard references, (Crane and Napier, 1989, Eq. 7-31) and (Walker, 1989, Eq. 9), so $\Delta S_{uw} = \sqrt{w_k} \Delta S_k$ for any k .

The remainder of this chapter is concerned with determining values of D_k and T_k that best balance the conflicting requirements of high resolution, low thermal noise, and immunity to various forms of deconvolution instabilities.

Note that when the data weights are statistical, subsequent processing programs should maintain this statistical nature whenever possible. Time averaging programs usually propagate the variances properly, and may offer several options to estimate the output variance after nonlinear processing or in the presence of nonthermal noise. But when combining multiple data sets, there is a distressing tendency to allow an arbitrary rescaling of the relative weights between the data sets. This practice is not recommended because it destroys the statistical nature of the weights and inevitably leads to a needless degradation of sensitivity. It arises as a crude method of controlling the total imaging weights, and is necessary only due to the inflexible nature of some contemporary mapping programs. The weights w_k should be regarded as a description of the data and not as a free parameter. Control of the imaging should be accomplished with D_k and T_k .

The second simplification we make in the noise analysis is to ignore the convolutional gridding step normally used in synthesis imaging, and described in Section 2.4. Even ignoring the aliased power, our simplified calculation is only correct at the map center where the transform of the convolution function is unity. Both of these effects are negligibly small in practice, however. The noise expressions derived in this chapter were spot checked by simulation over a wide variety of coverages and imaging parameters. Visibility data sets were constructed with thermal noise and no sources. The RMS value measured from the inner quarter of the dirty map was compared to the known amount of simulated thermal noise and the expected degradation due to the weighting. Typical agreement was found to be several parts in 10^4 , which in all cases was within the statistical variations expected from the simulation.

In the usual continuation of the noise analysis, statements similar to the following quotation are often made. “In the absence of strong sources, the noise at all points in the image have the same probability distribution” (Walker, 1989, p. 177) While true, this leads to the erroneous assumption that the variance of the dirty map is a function of source structure. To the extent that the aliasing due to convolutional gridding can be ignored, the latter assumption is not true. The Fourier transform is linear, and so the dirty image can be written as the transform of the true visibilities plus the transform of the complex noise.

$$I^D = \mathfrak{F}[W\mathcal{V}] + \mathfrak{F}[W\varepsilon]$$

The variance of the image is

$$\begin{aligned} (\Delta I^D)^2 &= \langle (I^D)^2 \rangle - \langle I^D \rangle^2 \\ &= \langle \mathfrak{F}^2[W\mathcal{V}] + 2\mathfrak{F}[W\mathcal{V}]\mathfrak{F}[W\varepsilon] + \mathfrak{F}^2[W\varepsilon] \rangle - \langle \mathfrak{F}[W\mathcal{V}] + \mathfrak{F}[W\varepsilon] \rangle^2 \end{aligned}$$

Since both the real and imaginary parts of the complex noise have zero mean, all of the $\langle \mathfrak{F}[W\varepsilon] \rangle$ terms vanish and this becomes

$$(\Delta I^D)^2 = \langle \mathfrak{F}^2[W\mathcal{V}] \rangle + \langle \mathfrak{F}^2[W\varepsilon] \rangle - \langle \mathfrak{F}^2[W\mathcal{V}] \rangle = \langle \mathfrak{F}^2[W\varepsilon] \rangle$$

Examination of equation 3.1 shows that each term of the sum is spatially variant only through the phasor. Since the probability density of the complex noise is circularly symmetric,

$$p(\text{Re } \varepsilon, \text{Im } \varepsilon) \propto \exp(-\text{Re}^2 \varepsilon / 2\sigma^2) \exp(-\text{Im}^2 \varepsilon / 2\sigma^2) = \exp(-|\varepsilon|^2 / 2\sigma^2),$$

any slice through this density via rotation by a phasor and taking the real part will result in the same Gaussian probability density, though not necessarily the same realization.³ Consequently the noise is the same at all points in the map and we can simply write the expression for the dirty map at the phase center,

$$I^D(0, 0) = 2C \sum_{k=1}^L T_k D_k w_k \text{Re}(\mathcal{V}_k + \varepsilon_k), \quad (3.2)$$

where we have combined the Hermitian conjugate pairs and now sum over L points. To make the units come out in terms of flux density,⁴ we normalize by setting

$$C \equiv 1 / \sqrt{2 \sum_{k=1}^L T_k D_k w_k}$$

Each term in equation 3.2 is statistically independent, so noting that $\langle \text{Re}^2 \varepsilon_k \rangle = \sigma_k^2$ we may simply combine the error terms in quadrature to obtain the standard expression for the RMS thermal noise in the dirty image.

$$\begin{aligned} \Delta I^D &= 2C \sqrt{\sum_{k=1}^L T_k^2 D_k^2 w_k^2 \sigma_k^2} \\ &= 2C \Delta S_{uw} \sqrt{\sum_{k=1}^L T_k^2 D_k^2 w_k} \\ &\propto \sqrt{\sum_{k=1}^L T_k^2 D_k^2 w_k} \end{aligned} \quad (3.3)$$

³This can be shown more formally either by convolving the probability densities of $\cos \theta \text{Re } \varepsilon$ & $\sin \theta \text{Im } \varepsilon$ or by integrating the joint probability density of $\text{Re } \varepsilon$ & $\text{Im } \varepsilon$ over $\text{Re } \varepsilon - \text{Im } \varepsilon$.

⁴The units of the dirty map are somewhat difficult. The definition we take here assures that a point source of flux density S has a peak numerical value S on the dirty map. Neither the integral of the point source response nor any property of an extended source response is rigorously defined. Practically speaking, if one integrates the dirty map of a compact source over a region somewhat larger than its support, one obtains a result similar to that of integrating the true source convolved with a Gaussian fitted to the main lobe of the (peak normalized) dirty beam. Hence the dirty image as normalized above is usually assigned the rather nebulously defined unit, “flux density per dirty beam”.

The last proportionality is with the understanding that the sum of the total weights are to be held constant, as is usually done. In Section 6.1, we will consider high-precision methods of estimating ΔS_{uw} directly from the visibility data, but for now we will simply assume it known to sufficient precision via a priori calibration.

3.3 Weighting as the Solution to a Minimization Problem

We are now in a position to derive the uniform and natural weights via a minimization formalism that is not widely known and which provides a motivation for the subsequent work.

3.3.1 Natural Weighting

Suppose we wish to find the weights D_k that minimize ΔI^D , subject to the constraint that the total weights sum to some constant, $W = 1/2C$. Equation 3.4 involves the weights both through the term under the radical, and via the “constant” C . The latter will cause the partial derivatives to become unpleasantly complicated, even though the total weight is constrained to W . Consequently, we minimize $(\Delta I^D/2C)^2$ and handle the constraint on the total weights with a Lagrangian multiplier as described in Appendix A. The same constraint ensures that the weights which minimize this problem also minimize ΔI^D . Holding the taper and statistical weights fixed, we write

$$\frac{\partial}{\partial D_k} \left[\left(\frac{\Delta I^D}{2C} \right)^2 + \lambda \left(\sum T_k D_k w_k - W \right) \right] = 0$$

$$2T_k^2 w_k^2 \sigma_k^2 D_k + T_k w_k \lambda = 0 \quad \forall k \quad \Rightarrow \quad T_k D_k w_k \sigma_k^2 = -\lambda/2 = \text{const}$$

This leads to a total weight

$$T_k D_k w_k \propto 1/\sigma_k^2$$

This is natural weighting, and is achieved by setting $T_k = D_k = 1$. As with a simple weighted mean, the very best one can do from a signal-to-noise standpoint is to weight all visibility samples with their inverse variance. This also leads to a simple normalization of the thermal noise. With natural weighting, equation 3.4 becomes

$$\begin{aligned} \Delta I_{nat}^D &= 2C_{nat} \sqrt{\sum_{k=1}^L w_k^2 \sigma_k^2} \\ &= \sqrt{\sum_{k=1}^L w_k^2 \sigma_k^2} / \sum_{j=1}^L w_j \quad (3.4) \\ &= \Delta S_{uw} / \sqrt{\sum_{k=1}^L w_k} \end{aligned}$$

Even without knowing the proportionality constant between w_k and $1/\sigma_k^2$, one can always form the quantity

$$\begin{aligned}\Delta I^D / \Delta I_{nat}^D &= 2C \sqrt{\sum_{k=1}^L T_k^2 D_k^2 w_k^2 \sigma_k^2} / 2C_{nat} \sqrt{\sum_{j=1}^L w_j^2 \sigma_j^2} \\ &= \sqrt{\sum_{k=1}^L T_k^2 D_k^2 w_k / \sum_{j=1}^L w_j} \left(\sum_{l=1}^L w_l / \sum_{i=1}^L T_i D_i w_i \right) \\ &= \sqrt{\left(\sum_{k=1}^L T_k^2 D_k^2 w_k \right) \left(\sum_{j=1}^L w_j \right)} / \sum_{i=1}^L T_i D_i w_i\end{aligned}\quad (3.5)$$

This is the factor by which an arbitrary weighting has degraded the thermal RMS noise from the optimum natural weighting. It is referred to as the normalized thermal RMS or the normalized point source sensitivity. Every good visibility mapping program should print this quantity, though few do.⁵ I believe that the surprising lack of progress in visibility weighting over the last two decades has been partly due to the lack of routine feedback about the thermal noise. Until the astronomer knows quantitatively what the weighting is doing to the thermal noise, he cannot make an informed choice between different options. If the quantity ΔS_{uw} is available, the program can print the expected image RMS in flux density units as well. For coding purposes, it may be easier to assume that ΔS_{uw} is known, setting it to unity if not. The normalized RMS may simply be calculated via two applications of equation 3.4, and the assumed value will divide out.

3.3.2 Sensitivity to Resolved Sources

We make a brief digression to calculate the optimum weighting for the detection of a resolved source, since the theoretical machinery for the calculation is already in place. We assume without loss of generality that the source has been phase shifted to the map center, and write the visibilities as \mathcal{V}' to indicate the presence of the phase shift. Divide equation 3.2 by equation 3.4 to obtain

$$\frac{I^D(0,0)}{\Delta I^D} = \frac{\sum_{j=1}^L T_j D_j w_j \operatorname{Re}(\mathcal{V}'_j)}{\left(\sum_{k=1}^L T_k^2 D_k^2 w_k^2 \sigma_k^2 \right)^{1/2}}\quad (3.6)$$

Consider a naturally weighted, untapered map so that $D_k = T_k = 1$. Further assume that all samples have common variance, σ^2 . The SNR for this case becomes

$$\frac{I^D(0,0)}{\Delta I^D} = \frac{\sum_{j=1}^L \operatorname{Re}(\mathcal{V}'_j)}{\sigma \sqrt{L}}$$

⁵The “Sum of gridding weights” reported by many *AIPS* tasks is proportional to $\sum T_k D_k w_k$. Unfortunately, this quantity cannot be used by itself to evaluate equation 3.6.

Clearly if the source is highly resolved and only a few samples have significant signal, the SNR actually goes *down* with increasing number of samples. Adding additional samples without signal merely increases the noise and obscures the signal from the good samples. Conversely, if all samples have a common flux density, S , then the above equation reduces to $\text{SNR} = S\sqrt{L}/\sigma$, and the SNR goes *up* with increasing number of samples.

What, then, is the proper way to combine samples of an arbitrary resolved source to maximize the SNR? Notice that there is no particular normalization required for the total weights. Multiplying all weights by a constant factor will produce exactly the same image and SNR. Thus we are free to chose any normalization that we like. Once again, we use a Lagrange multiplier and choose to hold constant the square of the denominator in equation 3.6. This fixes the normalization of the weights. Simultaneously, we maximize the numerator which has the net effect of maximizing the entire expression.

$$\frac{\partial}{\partial T_n} \left[\sum_{j=1}^L T_j D_j w_j \text{Re}(\mathcal{V}'_j) + \lambda \left(\sum_{k=1}^L T_k^2 D_k^2 w_k^2 \sigma_k^2 - T \right) \right] = 0$$

$$D_n w_n \text{Re}(\mathcal{V}'_n) + 2\lambda T_n D_n^2 w_n^2 \sigma_n^2 = 0$$

$$\frac{\text{Re}(\mathcal{V}'_n)}{T_n D_n w_n} = \frac{-\lambda}{2} w_n \sigma_n^2 = \text{const}$$

The maximum SNR for an arbitrary source at a given point is achieved when the total weight for each sample is proportional to the source visibility contribution to that point.

When the source is symmetric and reasonably⁶ centrally peaked, the dirty map will be as well, and $\text{Re}(\mathcal{V}'_j) = |\mathcal{V}'_j|$. The result becomes the simpler

The maximum SNR for a symmetric centrally peaked source is achieved when the total weight for each sample is proportional to the source amplitude.

The question of sensitivity to resolved sources is somewhat complicated. Crane & Napier (1989) give a discussion of sensitivity to extended sources and elect to use the SNR at a point as their measure. Cornwell (1984) advocates least squares fitting of a Gaussian to noise-corrupted visibility measurements of same, assuming the position and size of the source are known. The expected error in the fitted flux density is then used to define the sensitivity of the array to extended emission at that size scale. The resulting expression is similar to but quantitatively quite different from the image plane response at a point. In this chapter we will use the thermal noise (point source sensitivity) as the figure of merit, and simply note that the response to smooth emission will scale roughly as the square of the beam size.

⁶Certain pathological cases might take a centrally peaked source and produce a dirty map with a central depression.

3.3.3 Uniform Weighting

To explore uniform weighting, we consider a simple form of gridding; cell averaging. This consists of merely assigning each visibility to the nearest grid point and defining the gridded quantities by

$$\left. \begin{aligned} W_{pq} V_{pq} &= \sum_k T_k D_k w_k V_k \\ W_{pq} &= \sum_k T_k D_k w_k \end{aligned} \right\} \text{for } \begin{cases} |u_k - u_p| < \Delta u / 2 \\ |v_k - v_q| < \Delta v / 2 \end{cases} \quad (3.7)$$

The two expressions taken together define V_{pq} . For notational convenience, grid points are indexed by signed integers. $(0, 0)$ is the center of the grid so that $u_p = p\Delta u$ and $v_q = q\Delta v$. Translating these expressions into computer storage indices is a simple shift of origin. In later equations we will use the shorthand notation $k \in c_{pq}$ to mean all indices k with (u_k, v_k) in gridding cell (p, q) as written above. An indexed variable such as D_k appearing outside a summation over the same index is understood to be constant for all summed indices.

If the V_k in a given grid cell around (u_p, v_q) have a common expectation,⁷ \mathcal{V}_{pq} , then clearly $\langle V_{pq} \rangle = \mathcal{V}_{pq}$. Note also that if the D_k and T_k are roughly constant (as is usually the case), then the linear sum has exactly the form needed to minimize the variance of V_{pq} . This is why we can use the ungridded form of the imaging equation to calculate the map plane thermal noise. There is little SNR loss in gridding the visibilities — the major loss comes from combining the gridded visibilities with nonoptimal weights.

The gridded equivalent to the continuous mapping equation 3.1 is

$$I_{jk}^D = I^D(j\Delta\ell, k\Delta m) = C \sum_{\substack{p=-N_\ell/2+1 \\ -N_\ell/2+1}}^{N_\ell/2} \sum_{\substack{q=-N_m/2+1 \\ -N_m/2+1}}^{N_m/2} W_{pq} V_{pq} \exp(2\pi i(p\Delta u\ell + q\Delta v m)) \quad (3.8)$$

$$C = 1/W = 1/\sum_{p,q} W_{pq} = 1/2 \sum_k T_k D_k w_k$$

This has been written out over the entire Fourier plane, assuming each visibility has been gridded twice to include both conjugations. Since each ungridded visibility enters the sum through exactly two gridded weights, we recover the same normalization as equation 3.2. In more elaborate gridding schemes where the two forms of the normalization may not be exactly the same, the discrete equation is normalized with the sum of the gridded weights. The sum limits assume that the image size is even, though as shown in Section 2.5, an odd image size is often preferable. In such a case, the limits go from $-[N_\ell/2]$ to $[N_\ell/2]$. Neither

⁷Under the assumption that all visibilities in the cell have a common expectation, equation 3.7 is the best that one can do. If the visibilities vary significantly across a gridding cell (or across the support of the convolution function) and one had a perfect visibility model, there is a minimum variance sum with different coefficients than these. But then there is little need for data! It remains to be seen if an approximate model could be used without degrading the accuracy of the expectation. At best such a technique could only bring the thermal noise back down to that predicted by equation 3.4.

property affects our calculations here. Whatever the image size, we have that $\Delta u \Delta \ell = 1/N_\ell$ and $\Delta v \Delta m = 1/N_m$, so the exponential kernel becomes $\exp(2\pi i(jp/N_\ell + kq/N_m))$.

Suppose now that instead of minimizing the variance of the dirty map we decide that it is the sidelobes of the dirty beam which are important. The dirty beam, $B(\ell, m)$, is just the discrete transform of the gridded weights.

$$B_{jk} = B(j\Delta\ell, k\Delta m) = \frac{1}{W} \sum_{\substack{p= \\ -N_\ell/2+1}}^{N_\ell/2} \sum_{\substack{q= \\ -N_m/2+1}}^{N_m/2} W_{pq} \exp(2\pi i(jp/N_\ell + kq/N_m))$$

To minimize the sidelobes, we minimize the squared L_2 norm of B which is

$$\|B\|^2 = \sum_{\substack{j= \\ -N_\ell/2+1}}^{N_\ell/2} \sum_{\substack{k= \\ -N_m/2+1}}^{N_m/2} |B_{jk}|^2$$

By Parseval's theorem (Bracewell, 1986, p.369) this is proportional to the L_2 norm of the beam's discrete Fourier transform, the gridded weights.

$$\|B\|^2 = \frac{1}{N_\ell N_m W^2} \sum_{\substack{p= \\ -N_\ell/2+1}}^{N_\ell/2} \sum_{\substack{q= \\ -N_m/2+1}}^{N_m/2} |W_{pq}|^2$$

Once again we minimize a portion of the desired metric, and hold the remainder constant with a Lagrangian multiplier and normalization constraint. Also, not all W_{pq} are free parameters in the minimization. Gridding cells with no visibility points in them or zero total statistical weight are fixed at zero by equation 3.7. Hence we minimize only over the parameters which can vary.

$$\frac{\partial}{\partial W_{pq}} \left[N_\ell N_m W^2 \|B\|^2 + \lambda \left(\sum_{p,q} W_{pq} - W \right) \right] = 0$$

$$2W_{pq} + \lambda = 0 \quad \forall p, q \quad \Rightarrow \quad W_{pq} = -\lambda/2 = \text{const}$$

Usually the density weight is calculated before the taper, so we will set $T_k = 1$ here. Since

$$W_{pq} = \sum_k T_k D_k w_k \quad \text{for } (u_k, v_k) \text{ in grid cell } (p, q),$$

we see that a constant D_k for all visibility points in the same cell will satisfy the minimization.

$$D_k = \begin{cases} \text{const} / \sum_{k \in c_{pq}} w_k & \sum_{k \in c_{pq}} w_k > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.9)$$

This is uniform weighting, the name coming from the fact that the gridded weights are as uniform as is possible. Since most u - v samplings tend to have the densest coverage near the origin, the net effect of uniform weighting is to upweight the outer visibilities relative to

those closer to the origin. Uniform beams are characterized by tighter main lobes (higher resolution) and poorer thermal noise than their naturally weighted counterparts.

The property $W_{pq} = \{\text{const or } 0\}$ is responsible for the other common definition of uniform weighting. A uniform beam is one which is a multiple of itself after self-convolution. Since self-convolution is merely a squaring of the gridded weights in the Fourier plane, clearly $W_{pq}^2 = \{\text{const}^2 \text{ or } 0\}$ satisfies the uniform minimization criterion and is also a multiple of the original beam. Any beam derived from two or more unequal nonzero gridded weights will not scale in this way so the two definitions are equivalent.

Some tasks in \mathcal{AIPS} are at odds with the literature and do not use these definitions of “uniform weighting”. D_k is set to the inverse of the *number* of samples in the cell, rather than the inverse of the snr weight sum. Since all the visibilities in a gridding cell are multiplied by a constant in either approach, and the visibilities are still combined similarly to equation 3.7, there is no undue loss in sensitivity even when the snr weights are not equal. But in this number count uniform weighting the gridded weights are not constant unless all the snr weights are equal. As of the 15JAN95 release, **MX**, **UVMAP**, and **HORUS** all use number count uniform weighting. The tasks **WFCLN** and **SCMAP** use the weight sum uniform weighting as described here. The imaging tasks in \mathcal{AIPS} are scheduled for an overhaul, so both styles of uniform weighting will probably becomes available to all imaging tasks in future versions. The task **IMAGR** has appeared in the 15JUL95 release of \mathcal{AIPS} and is extremely flexible in terms of weighting and other features. It seems likely to become the task of choice for most mapping and CLEAN based deconvolution in \mathcal{AIPS} .

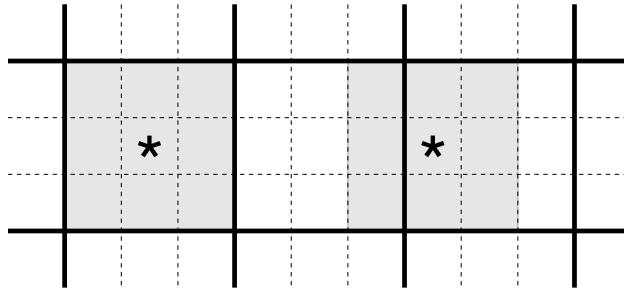
3.3.4 Super-Uniform Weighting

Since the peak of the dirty beam is normalized to unity, minimizing $\|B\|^2$ will favor both a tighter main lobe and smaller mean square sidelobes in the rest of the beam. These requirements often conflict, with a tighter main lobe leading to higher sidelobes. The relative importance of these two criteria is influenced by the number of pixels in each category. Minimizing over a smaller area of the map means that a larger fraction of the pixels in the sum come from the main lobe. This leads to a smaller fitted beamwidth than uniform weighting, at the expense of higher sidelobes. (Both uniform and super-uniform weighting will almost certainly have smaller mean squared sidelobes than natural weighting.) Consequently, some control over the tradeoff between beamwidth and far out sidelobes can be had by varying the fraction of the map over which $\|B\|^2$ is minimized.

Suppose that we wish to minimize $\|B\|^2$ over a box around the origin with sides FOV_w times those of the original image. (FOV_w is called the weighting field of view.) We can no longer use Parseval’s theorem to connect this region to the gridded weights. However, consider a second map exactly the size of the region we wish to minimize over, with the same pixel spacing as the original image. The uniform density weights derived for this second image are exactly what we desire, which can then be applied to an image

of any size. The pixel spacing in the gridded Fourier plane is $\Delta u = 1/N_\ell \Delta \ell$, and N_ℓ is a fraction FOV_w smaller than before. Hence if we wish to minimize over a linear fraction FOV_w of the original mapsize, we need only calculate the uniform weights over a box in the visibility plane that is $1/FOV_w$ times larger than the normal uniform weight box. This can be done simply by multiplying the u - v coordinates of each sample by FOV_w during the reweighting process and using the normal uniform reweighting grid. So long as the visibility samples fit onto the grid, this approach allows both super-uniform weighting ($FOV_w < 1$) and sub-uniform weighting ($FOV_w > 1$). This approach is used in SDE and some tasks within \mathcal{AIPS} .

There is nothing particularly special about the region over which sidelobes are minimized. The particular choice of the size of the map is essentially for convenience and leads to a simple form of reweighting in the u - v plane. But that should not deter the astronomer from making a different choice if it leads to better beam properties for a given application.



Again there are some implementation differences between the pure mathematical formulation of super-uniform weighting and how it is implemented in \mathcal{AIPS} . Some tasks implement super-uniform weighting by summing the visibility weights (or counting the visibilities) in an adjacent number of normally sized weighting cells. Here we have a situation where the \mathcal{AIPS} parameter `UVBOX` = 1, which implies a 3×3 cell uvbox. (The box size in cells is $2\text{UVBOX} + 1$.) The corresponding SDE case is specified as $FOV_w = 1/3$. The two approaches are not quite equivalent. If the visibility to be reweighted is found in the center third of the large reweighting box, as in the left example, both the scaled cell and integer box approach will consider all visibilities in the shaded region for the reweighting. In the case of the right hand example, the target visibility is towards the edge of a large scaled cell. The scaled cell approach will consider all visibilities in the large box delimited by the heavy lines. The integer box approach will consider visibilities in the shaded box. That is, the integer box approach will more closely center the reweighting box on the visibility. There are merits to both approaches. The scaled cell approach is more flexible in that there is no need to constrain FOV_w to any particular function of an integer. This is particularly useful when one wishes to change the size of an image and hold the weighting constant. The scaled cell approach is the result of a analytical minimization, and hence more easily interpreted. The integer box approach has the virtue of being less susceptible to irregularities of how the points fall on the reweighting grid. We consider a generalization of the integer box

approach in Section 3.5.1 for this very reason. As of the 15JAN95 release of *AIPS* the tasks `MX`, `UVMAP` and `HORUS` use the integer box approach when processing `XY` ordered data. `MX` and `HORUS` use the scaled cell approach when processing non-`XY` ordered data. The tasks `WFCLN` and `SCMAP` always use the scaled cell approach. Unfortunately, even when using the scaled cell approach, *AIPS* specifies the uvbox size with the integer `UVBOX` argument — it is not possible to sub-uniform weight in *AIPS* at the moment. Again, the imaging tasks in *AIPS* are scheduled for an overhaul, and it is likely that all will become more flexible in their weighting options.

3.4 Beamwidth and Thermal Noise

We can now explore a point made in Section 2.6, that beamwidth and sensitivity is not a unique function of the sampling and type of weighting. Especially with uniform weighting, it can vary quite appreciably when modifying “innocuous” parameters like the image size. The major effect is that the uniform reweighting cell size is the inverse of the physical field of view, as stated above. So if one were to decrease the size of a uniformly weighted map by a linear factor of three, the effect on the weighting is the same as specifying $FOV = 1/3$ within `SDE`. In Table 3.1 there is an example of a fitted beamwidth being $.156''$ with a cell size of $.0667''$ and an image size of 32^2 . By simply increasing the image size to 2048^2 , the fitted beam size rises to $.247''$ — a considerable change. The normalized thermal noise varies from 3.99 to 1.23 over the same range. The u - v coverage for this particular observation (VLA Full Track (3C48), given as the upper right panel in Figure 3.12a) is fairly circular with $u_{max} = 935 \text{ k}\lambda$ and $v_{max} = 933 \text{ k}\lambda$. Thus either \hat{u}_{max}/u_{max} or \hat{v}_{max}/v_{max} is reasonable as an oversampling figure of merit. The geometric mean of these is tabulated, here and in later tables.

The beamwidth in these tables is determined by a nonlinear fit of an elliptical Gaussian to the dirty beam pixels as described in Appendix C. The geometric mean of B_{maj} and B_{min} is tabulated. In general, the nonlinear beamfit is used in numerical plots and tables in this dissertation. Images have generally been restored with a linear fitted Gaussian, simply because this happens to be the software default. The difference between the two algorithms is typically about a percent in mean beamwidth. In general, the nonlinear beamfit is recommended, as it should have the highest photometric integrity, but for most projects it makes little difference.

As expected, both increasing the image size while holding cell size fixed and increasing the cell size while holding image size fixed lead to a larger field view, less super-uniform weighting, a larger main lobe and lower thermal noise. To think about this in the Fourier plane, consider a gridding cell near the origin with two points in it. Since there are usually more points near the origin, these multipoint gridding cells will preferentially also appear near the origin. If we subdivide this cell into two parts and each visibility falls into a different part, what was one term in the Fourier Sum for the dirty image becomes

points per beam	cellsize (arcsec)	over- sampling	Image Size (pixels)					
			32	64	128	256	512	1024
10	.0200	5.52	.144	.151	.155	.167	.183	.202
8	.0250	4.42	.150	.152	.159	.172	.188	.207
6	.0333	3.32	.148	.156	.163	.178	.196	.215
5	.0400	2.76	.150	.154	.167	.184	.202	.220
4	.0500	2.21	.154	.162	.175	.187	.207	.224
3.5	.0571	1.93	.155	.162	.177	.194	.210	.226
3	.0667	1.66	.156	.166	.178	.198	.216	.231
2.5	.0800	1.38	.156	.168	.185	.201	.219	.234
2	.100	1.10	***	***	.187	.207	.224	.238
1.5	.133	0.83	***	***	***	***	.246	.270

Table 3.1: Uniform Beamwidth in arcseconds vs. “innocuous” imaging parameters. The first column is an approximate points per beam, relative to a nominal beamwidth of $.2''$. Oversampling is the geometric mean of \hat{u}_{max}/u_{max} and \hat{v}_{max}/v_{max} . The data set is the “VLA Full Tracks (3C48)” case of Figure 3.12a. Cases where the beam fit has failed are marked with asterisks.

two. Uniform weighting assures that each term in the sum contributes equally. A cell further out with but one visibility in it will remain one term in the sum. Thus as the size of the gridding cell decreases, the contribution of the lower spatial frequencies and the beamwidth increases. There are clearly second order effects in Table 3.1. Compare beamwidth at a cellsize and image size with that at twice the cellsize and half the image size. At reasonably large image sizes they agree well, but there is also a slight trend towards smaller images having larger than expected fitted beamwidths. This is probably due to the fitting algorithms using a maximum of 11×11 points as input to the fit. It may also reflect convolutional gridding problems in the smallest images. The lowest line of the table has the highest spatial frequency points being dropped off the edge of the transform grid, so the beamwidth naturally increases.

The naturally weighted maps show much less variation in fitted beamwidth against image parameters. The slight trend of larger than expected fitted beam widths for small images and large cell size is still present, and in this table the last three rows are affected by data falling off the edge of the transform grid. The very slight variations with cellsize but not image size probably reflect variations in the fit due to the same continuous function being sampled at different point.

Finally, we have a tabulation of normalized thermal noise against image parameters for the uniform case. Again as expected, smaller beamwidth is highly correlated with higher thermal noise. The most extreme examples are obviously in regions of parameter space rarely used, but there is still considerable variation across more common parameters. The tradeoff is made much clearer in Figure 3.1 with a scatter plot of beamwidth against thermal noise. We will see the general shape of this tradeoff curve many times in this chap-

points per beam	cellsize (arcsec)	over- sampling	Image Size (pixels)					
			32	64	128	256	512	1024
10	.0309	3.57	0.309	0.309	0.309	0.309	0.309	0.309
8	.0386	2.86	0.311	0.311	0.311	0.311	0.311	0.311
6	.0515	2.14	0.311	0.311	0.311	0.311	0.311	0.311
5	.0618	1.79	0.310	0.310	0.310	0.310	0.310	0.310
4	.0773	1.43	0.309	0.309	0.309	0.309	0.309	0.309
3.5	.0883	1.25	0.312	0.311	0.311	0.311	0.311	0.311
3	.103	1.07	0.315	0.310	0.309	0.309	0.309	0.309
2.5	.124	0.89	0.334	0.318	0.314	0.314	0.313	0.313
2	.155	0.71	0.374	0.333	0.322	0.319	0.318	0.317
1.5	.206	0.54	0.448	0.387	***	***	***	***

Table 3.2: Same quantities as Table 3.1 but with natural weighting instead of uniform, and a nominal beamwidth of $.309''$.

ter. Most of the processing parameters are interrelated, and varying one while holding the rest constant will typically produce this kind of curve.

points per beam	cellsize (arcsec)	over- sampling	Image Size (pixels)					
			32	64	128	256	512	1024
10	.0200	5.52	9.691	4.951	3.650	3.022	2.369	1.883
8	.0250	4.42	6.755	4.447	3.269	3.110	2.197	1.788
6	.0333	3.32	6.329	4.644	3.709	2.526	2.002	1.655
5	.0400	2.76	4.951	3.650	3.022	2.369	1.883	1.591
4	.0500	2.21	4.447	3.269	3.110	2.197	1.788	1.510
3.5	.0571	1.93	4.804	3.210	2.824	2.136	1.728	1.471
3	.0667	1.66	3.986	3.701	2.547	2.017	1.660	1.419
2.5	.0800	1.38	3.650	3.022	2.370	1.883	1.591	1.366
2	.100	1.10	3.269	3.110	2.197	1.788	1.510	1.299
1.5	.133	0.83	10.224	10.526	10.500	9.947	9.303	8.485

Table 3.3: Normalized RMS thermal noise vs. image size and cell size for uniform weighting. All other quantities are as Table 3.1. Italicized numbers represent the cases where the beam fit has failed.

3.5 Catastrophic Gridding Errors

The usual approach to implementing uniform weighting, used in both *AIPS* and *SDE*, is to grid the visibility weights on the same grid as will later be used to the grid the visibilities themselves. A second pass is made through the visibility data, and each weight is multiplied by the inverse of the sum of the number of the weights in that grid cell. Thus the sum of all modified weights in every occupied cell will be unity. Super-Uniform weighting is implemented by using a scaled grid for the weights or by counting up weights in adjacent

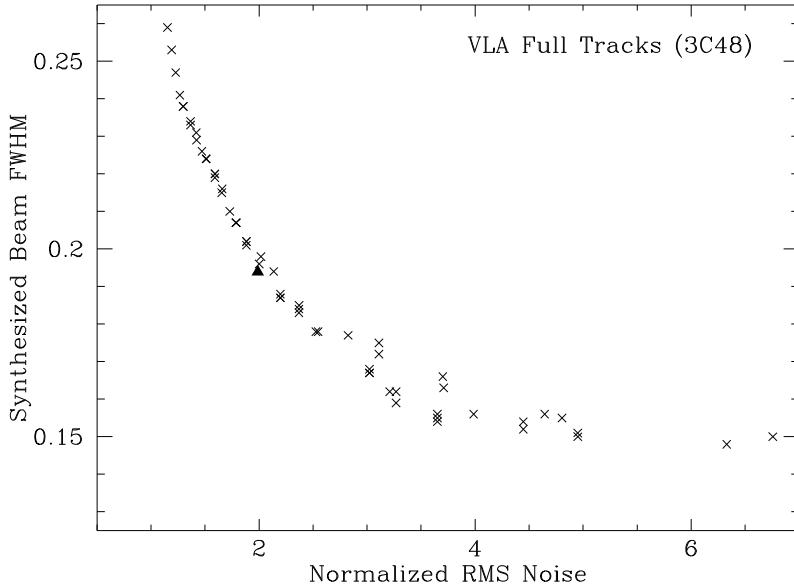


Figure 3.1: Beamwidth from Table 3.1 plotted against Thermal noise from Table 3.3. Note that some points are off scale. The solid triangle is the point representing the “standard” case of Figure 3.14.

cells on the standard grid. All of these approaches suffers from the instability that the local density of points is very susceptible to slight variations in the geometry of the u - v plane sampling tracks and the grid. Consider the case found in Figure 3.2. The samples are taken along an elliptical track, and all snr weights are presumed equal. There is no significant difference in the local density of u - v points and a uniform weighting algorithm should weight them all equally. However, in this case the u - v track has barely clipped one corner of a grid cell, and a single visibility will be assigned a weight much larger than the visibilities in adjacent cells where many visibilities have been gridded together. The thermal noise in each gridded visibility scales as the inverse square root of the number of samples, so gridded visibilities of quite different thermal noise levels will contribute equally to the final map. More to the point, equal noise levels in ungridded visibilities will contribute very *unequally* to the final map. In extreme cases similar to the one shown, normal thermal variation in a single visibility can be greatly magnified by the weighting and cause striping in the dirty map. Since the striping is actually in the data, the deconvolution algorithms will try faithfully to fit it, and the striping will be transferred to the final map. Even in cases where the gridding does not fail catastrophically, the drastic departure from the optimum statistically weighted sum can degrade the sensitivity of the map considerably.

Until this effect is appreciated, catastrophic gridding errors can be difficult to diagnose. When the dirty map is back transformed to the Fourier plane, the error will appear to be a single point of interference on a single baseline. A correlator glitch, perhaps. A plot of the ungridded visibilities will not show any particularly errant points, however.

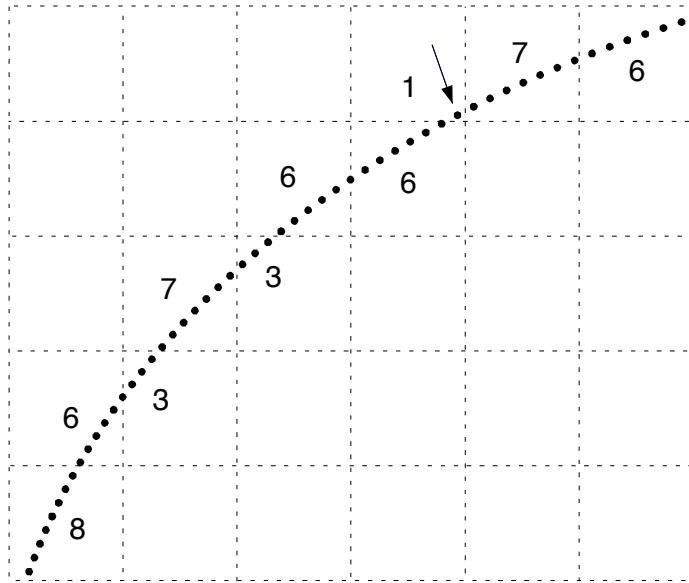


Figure 3.2: u - v tracks on a grid, demonstrating catastrophic gridding error. The numbers are the occupancy of each filled cell, and the weights in later mapping will be the inverse of these. The visibility marked with an arrow will weighted much more heavily than adjacent cells and this effect may cause striping in both the dirty and deconvolved image.

Nor will a back transform of the dirty beam show any unduly high gridded weights. The key symptom is that the striping will change when the field of view is altered, often in a dramatic manner after very small changes. This is similar to testing an image feature for internal aliasing by changing the field of view, but different in that aliased features will only change in a regular manner. This is the first example known to me of a ‘chaotic’ error dependence on incidental imaging parameters.

Figure 3.3 shows an example of catastrophic gridding error and some later solutions. The source is SN1987A, and the aim of the project was the highest resolution imaging possible. As a consequence, the image was heavily oversampled with $\hat{u}_{max}/u_{max} \approx \hat{v}_{max}/v_{max} \approx 6$. $\Delta\ell = \Delta m = .1''$, and the dirty image size is 512^2 . The data was kept at the highest time resolution available, 10 seconds, and the image was super-uniformly weighted with the scaled cell approach and $FOV_w = 1/4$. All of these factors conspire to create a situation where the size of the gridding box in the Fourier plane is large, and where there is a high density of low weighted samples along each track. This is exactly the kind of project most likely to encounter the problem. The other likely scenario is high fidelity mapping where we oversample to avoid representation problems. In such a project striping problems will be less severe than in this example, but the required fidelity of the final image will be higher as well.

In Figure 3.3, the striping is clearly visible. A logarithmic grey scale has been used, but the striping is visible on a linear transfer function as well. It is difficult to measure

quantitatively the amplitude of the striping in the presence of the beam sidelobes and thermal noise, but a crude attempt was made by differencing in quadrature the off-source RMS in the dirty map with the off-source RMS of a dirty model and measured thermal RMS from the internal scatter of the visibilities. The remaining contribution should be that of the stripes. This procedure yields

$$RMS_{stripes}^2 \approx RMS_{map}^2 - RMS_{model}^2 - RMS_{noise}^2 \approx 79^2 - 57^2 - 37^2 \approx 40^2 \mu\text{Jy}$$

This suggests that the RMS striping in the dirty map is about equal to that of the thermal noise. But of course the striping is completely coherent and hence highly visible. This estimation procedure breaks down when used on the deconvolved map, as the measured RMS is only $39 \mu\text{Jy}$. $39^2 - 37^2 \approx 12^2$, but with the uncertainties in measurement and how the noise has been modified, about all one can say is that the striping appears somewhat less relative to the noise, but still significant.

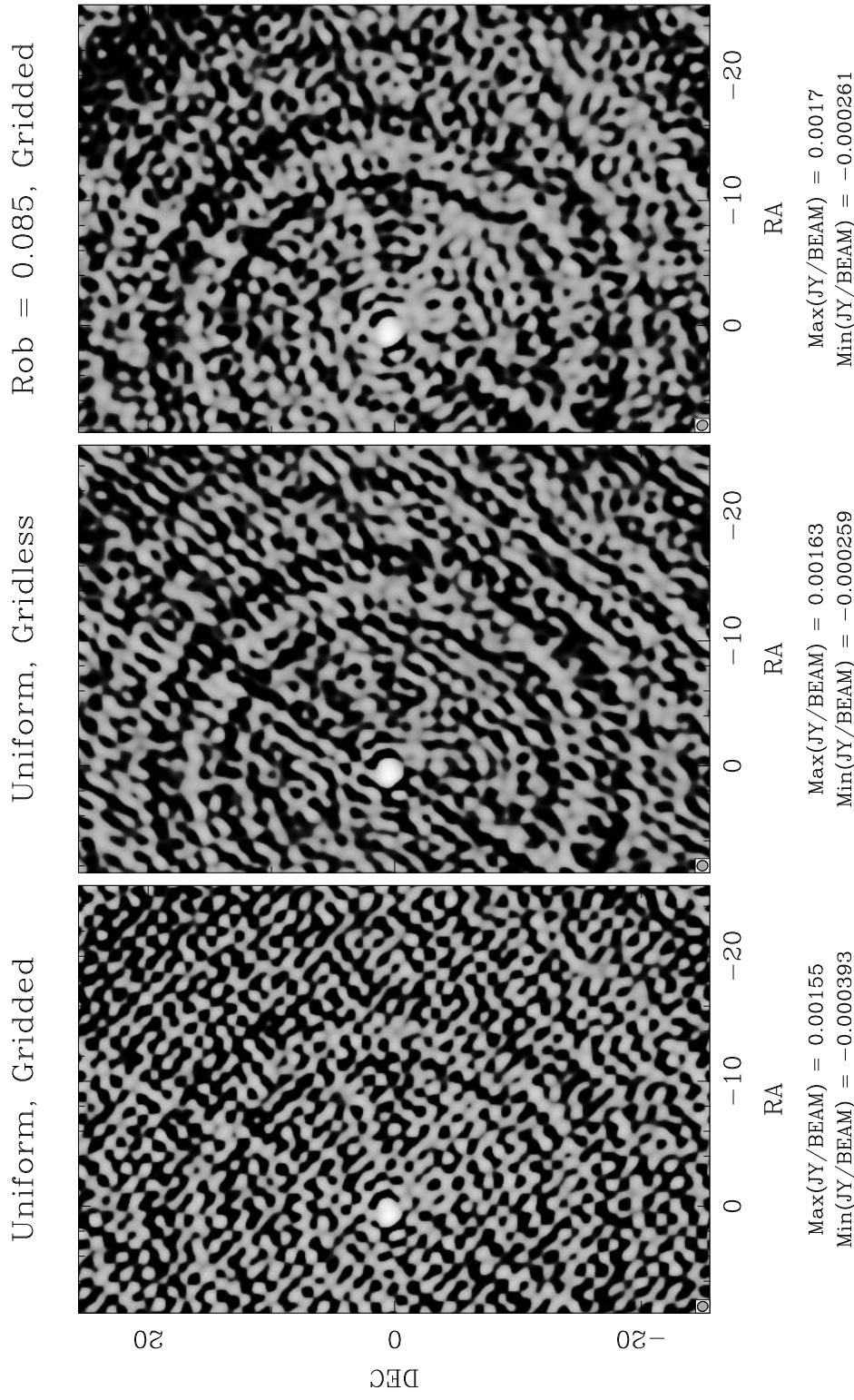


Figure 3.3: Stripping in the dirty map. See text for data details. Left panel is standard super-uniform weighting with normalized RMS noise & beamwidth of $(2.24/.815'')$. The middle panel uses the gridless weighting of Appendix E. $(1.98/.852'')$. The right panel again uses the gridded algorithm, but adds the robust weighting from Section 3.6. $(1.34/.852'')$. All panels use a common greyscale which is symmetrically logarithmic around zero.

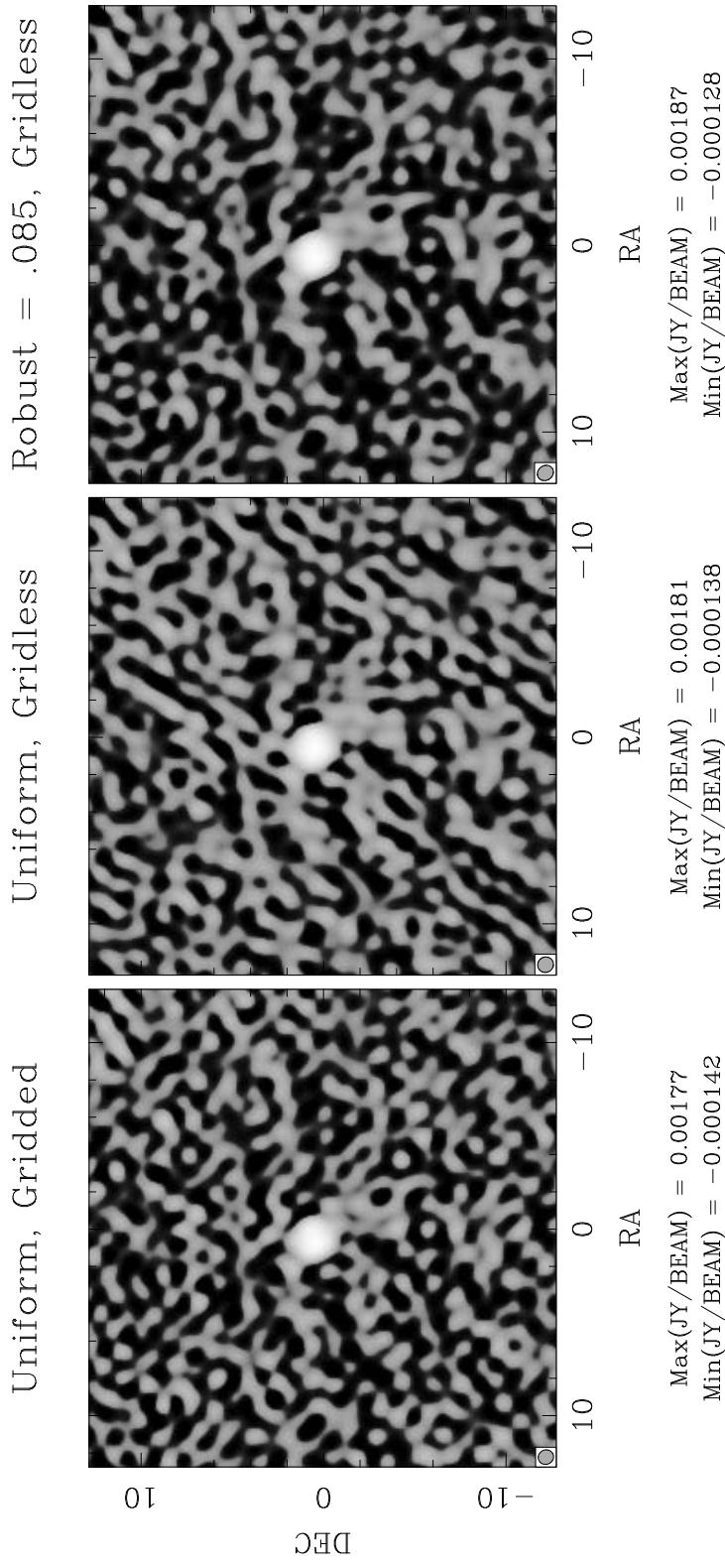


Figure 3.4: Stripping in the restored image. Maximum Entropy deconvolutions of the maps from Figure 3.3. The stripping is entering the restored image both from the deconvolution component model and from the residuals.

3.5.1 Gridless Weighting

The ubiquitousness of gridded weighting is primarily for computational reasons. It's true that Section 3.3.3 shows that gridded uniform weighting is the unique solution to the minimization problem on the discrete dirty beam. The gridded nature of the transform reflects itself in the gridded nature of the optimum weights. But there seems good use for a "uniform like" weighting that maintains the uniform density so far as possible, maximizes the sensitivity by treating equally sensitive points equally where possible, and that is not susceptible to catastrophic gridding errors. The obvious way to do this is to modify the definition of local density to a region actually centered on each individual visibility.⁸ Arguably, this is closer to the true local density than the approximation used in gridded weighting. With an elliptical region, the uniform density weights analogous to equation 3.9 are simply

$$D_k = \text{const} / \sum_j w_j \quad \text{for } \frac{(u_j - u_k)^2}{r_u^2} + \frac{(v_j - v_k)^2}{r_v^2} \leq 1 \quad (3.10)$$

where $r_u = \pi^{-1/2} \Delta u / \text{FOV}_w$ and similarly for r_v . The factor of $\pi^{-1/2}$ is so that the area of the consideration region is the same for both the gridded and gridless algorithms.

In general, computing the centered local density of visibility points is an $O(N^2)$ problem in the number of visibilities. With typical data sets in the hundreds of thousands of points or more, this is a daunting task. Presorting the visibilities in u or v will reduce the running time substantially, but the problem is still $O(N^2)$, and the proportionality constant degrades badly with decreasing FOV_w .

In Appendix E an algorithm is presented that calculates the centered local density of visibilities, using the definition above. It completely eliminates the problem with visibility tracks nicking gridding boxes. It is conceptually quite simple and might prove useful for other applications involving testing all pairs of visibilities, such as calibration using crossing points. It is still an $O(N^2)$ algorithm, though the proportionality constant is very good and running times for most contemporary problems are comparable to other stages of the mapping/deconvolution process. The algorithm is most suited to problems where the entire visibility data set and intermediate arrays can fit into virtual memory, but a disk based implementation would be possible. Running time does degrade substantially with weighting field of view, but not as badly as in simpler algorithms. Since distance comparisons are made between individual visibility points, we are able to use a circular neighborhood around each point rather than the rectangular neighborhood used in the usual algorithms. This removes at least one potential source of grid orientation bias on the sky. It is probable that a more sophisticated algorithm could be devised with better asymptotic performance, but the fast leading constant makes this algorithm quite adequate for the near future and possibly among the better solutions for contemporary problems.

⁸For this reason, I consider the gridless algorithm to be a relative to the integer box method of super-uniform weighting. Both make an effort to center the visibility in the box so far as is possible.

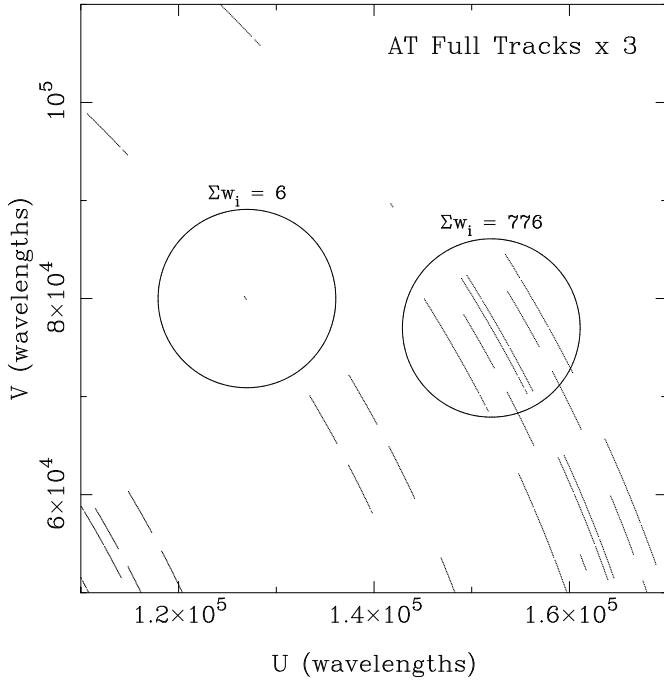


Figure 3.5: Isolated u - v sampling points. The circles are the regions in the u - v plane used to reweight two representative points. The sum of the weights within the two regions is 6 and 776 as labeled.

When the gridless weighting was used on the problem described in Section 3.5, the middle panel of figures 3.3 and 3.4 was obtained. The primary striping pattern has vanished, and the level of the striping has gone down somewhat, but a new set of stripes has been uncovered beneath the one present before. Examination of the data revealed a pathology in this particular data set. The full u - v sampling is given in the upper left panel of Figure 3.12b, AT Full Tracks (SN1987A). An extreme magnification of that sampling is given in Figure 3.5. This reveals several isolated u - v sampling points that have probably been produced by data editing. The points in the circle to the left will be added into the dirty map with weight $776/6 \sim 130$ times the weight of points within the densely sampled region to the right. In the case of an east-west array like the Australia Telescope, the problem is worse still in that anomalously reweighted points tend to occur at the same position angle. The cluster of three points not circled is also contributing to the striping.

The problem is very similar to that of the catastrophic gridding error discussed in the previous section. A few visibility points are being weighted anomalously high compared to others in the same region of the u - v plane. This reweighting is producing stripes in the map, due to normal thermal noise in the data.⁹ The difference is that this reweighting

⁹If it is a mechanical problem with the telescopes that is producing the isolated data, say loss of LO lock

anomaly is fundamentally due to the geometry of the u - v sampling, and not the particular reweighting grid used in the imaging step. The gridless weighting is clearly an improvement with regards to the normal reweighting and striping, but there are pathological cases like this one where it too will fail.

3.6 Robust Uniform Weighting

The striping in the middle panel of Figure 3.3 could of course be eliminated by simply editing out the isolated data, but of all the visibilities in the data set these are the most valuable, telling us about the source in an otherwise unmeasured region. Clearly the fix should be done in software, and the isolated visibilities should be included in the dirty image at some reasonable weighting level. Since the problem with uniform weighting is the magnification of thermal noise in isolated visibilities, the definition of “reasonable” in this context must include knowledge about the thermal noise in the visibilities. The reweighting algorithm should ‘know’ that it is allowed to upweight isolated points of high sensitivity more than isolated points of low sensitivity.

Uniform weighting is explicitly designed to remove all knowledge of the thermal noise from the gridded visibility. After the data has been gridded, all occupied visibility cells have the same weight, and there is no way to tell if that data is of high or low quality. By contrast, natural weighting is exactly the opposite. All possible information about the sensitivity of the original data is preserved in the gridded visibilities. But the shape of the synthesized beam is rarely what we desire with natural weighting. We are motivated to seek a weighting that combines some features of both uniform and natural weighting, maximizing both the sensitivity and the beam shape while balancing the conflicting requirements against each other in a reasonable way.

Natural weighting knows about the thermal noise. Uniform weighting knows about the dirty beam. We now apply the expected minimum L_2 norm criterion to a noise-corrupted dirty beam — the dirty image of a point source. If this fictitious point source strength is large compared to the thermal noise in the real data, the dirty map is identical to the dirty beam and we recover uniform weighting. If the source strength is negligible compared to the noise, we are minimizing the RMS noise per pixel and recover natural weighting. The strength of the point source is a smooth knob that can be varied, and at intermediate values we anticipate a beam with some compromise properties of both weightings. The weights derived from this criterion are then applied to the actual visibility data.

or tracking, then the points which just survive filtering on this basis might be prone to abnormally high statistical variations, exacerbating the striping problem.

For the initial portion of the analysis, we will start with the gridded visibilities and assume that the thermal noise in each visibility is known, with $\sigma_{pq}^2 = \langle \text{Re}^2 \varepsilon_{pq} \rangle = \langle \text{Im}^2 \varepsilon_{pq} \rangle$. The ‘source’ visibility is $V_{pq} = S + \varepsilon_{pq}$. Again invoking Parseval’s theorem

$$\begin{aligned}
\langle \|I^D\|^2 \rangle &= \left\langle \sum_{j,k} |I_{jk}^D|^2 \right\rangle \\
&= \left\langle \frac{1}{N_\ell N_m W^2} \sum_{p,q} |W_{pq} V_{pq}|^2 \right\rangle \\
&= \frac{1}{N_\ell N_m W^2} \sum_{p,q} \langle W_{pq}^2 |S + \text{Re } \varepsilon_{pq} + i \text{Im } \varepsilon_{pq}|^2 \rangle \\
&= \frac{1}{N_\ell N_m W^2} \sum_{p,q} W_{pq}^2 \langle (S + \text{Re } \varepsilon_{pq})^2 + \text{Im}^2 \varepsilon_{pq} \rangle \\
&= \frac{1}{N_\ell N_m W^2} \sum_{p,q} W_{pq}^2 \langle S^2 + 2S \text{Re } \varepsilon_{pq} + \text{Re}^2 \varepsilon_{pq} + \text{Im}^2 \varepsilon_{pq} \rangle \\
&= \frac{1}{N_\ell N_m W^2} \sum_{p,q} W_{pq}^2 (S^2 + 2\sigma_{pq}^2)
\end{aligned} \tag{3.11}$$

Once again we minimize a portion of the metric and hold the remainder constant with the normalization of the weights.¹⁰ And once again, we are minimizing only over the gridded weights that can vary, the rest being fixed at zero.

$$\begin{aligned}
\frac{\partial}{\partial W_{pq}} \left[N_\ell N_m W^2 \|I^D\|^2 + \lambda \left(\sum_{p,q} W_{pq} - W \right) \right] &= 0 \\
2W_{pq}(S^2 + 2\sigma_{pq}^2) + \lambda &= 0 \quad \forall p, q \quad \Rightarrow \quad W_{pq}(S^2 + 2\sigma_{pq}^2) = -\lambda/2 = \text{const} \\
W_{pq} &\propto \frac{1}{S^2 + 2\sigma_{pq}^2}
\end{aligned} \tag{3.12}$$

This is the essence of robust weighting. If $S \gg \sigma_{pq}$, then the dirty image we are minimizing over looks very much like the dirty beam. W_{pq} approaches a constant and we recover uniform weighting. If $S \ll \sigma_{pq}$, then we are essentially minimizing over an image of random noise, $W_{pq} \rightarrow \text{const}/\sigma_{pq}^2$ and we recover natural weighting.

Recall from equation 3.7 that the gridded visibilities are defined as

$$\begin{aligned}
V_{pq} &= \sum_{k \in c_{pq}} T_k D_k w_k V_k / \sum_{j \in c_{pq}} T_j D_j w_j \\
&= \sum_{k \in c_{pq}} T_k D_k w_k (S + \varepsilon_k) / \sum_{j \in c_{pq}} T_j D_j w_j \\
&= S + \sum_{k \in c_{pq}} T_k D_k w_k \varepsilon_k / \sum_{j \in c_{pq}} T_j D_j w_j
\end{aligned}$$

¹⁰Gridded or ungridded weights makes no difference here, since the sum of each is the same as that of the other.

Each term in the numerator sum is a complex Gaussian random variable with independent real and imaginary parts, each with variance $(T_k D_k w_k \sigma_k)^2$. The real and imaginary parts of each term will add in quadrature, so

$$\sigma_{pq}^2 = \sum_{k \in c_{pq}} (T_k D_k w_k \sigma_k)^2 / \left(\sum_{j \in c_{pq}} T_j D_j w_j \right)^2$$

Again, we assume that the taper is done after the density weighting, so that $T_k = 1$, and we assume that the density weights are the same for all visibilities in a gridding cell.¹¹.

$$\begin{aligned} \sigma_{pq}^2 &= \sum_{k \in c_{pq}} (w_k \sigma_k)^2 / \left(\sum_{j \in c_{pq}} w_j \right)^2 \\ &= \Delta S_{uw}^2 / \sum_{j \in c_{pq}} w_j \end{aligned}$$

Substituting this expression for σ_{pq}^2 into equation 3.12

$$W_{pq} = \sum_{k \in c_{pq}} T_k D_k w_k = D_k \sum_{k \in c_{pq}} w_k \propto \frac{1}{S^2 + 2\sigma_{pq}^2}$$

yields the form:

$$D_k = \begin{cases} \text{const} / \left(S^2 \sum_{k \in c_{pq}} w_k + 2\Delta S_{uw}^2 \right) & \sum_{k \in c_{pq}} w_k > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.13)$$

Clearly when $S = 0$ this recovers the natural weighting $D_k = \text{const}$ and when $S \rightarrow \infty$ we recover uniform weighting $D_k \propto 1 / \sum_k w_k$.

3.6.1 Robust Weighting and Wiener Filtering

There is an interesting cross over to conventional image restoration theory. The density weighting can be considered a filter acting on the naturally weighted dirty map and beam. Viewed in this way, the functional form of robust weighting clearly resembles the traditional Wiener filter, (Wiener, 1942). There is a generalization of the Wiener filter, the parametric Wiener filter (Andrews and Hunt, 1977, p. 150), which resembles robust weighting even more in that it has an adjustable parameter which plays the same role as the robustness here. The parametric Wiener filter falls into a family called Constrained Least-Squares Filters, which minimize some linear operator on the estimated image subject to some additional constraints. If one chooses to minimize the effective noise to signal ratio of the robust weighted dirty map while simultaneously constraining the norm difference

¹¹Obviously, this is an approximation for the gridless weighting described in Section 3.5.1 It will still yield a reasonable weighting, if perhaps not precisely the one that minimizes equation 3.11.

between the natural and robust dirty maps to a fixed value, the result is a form of the parametric Wiener filter very close to Robust weighting. In Fourier space, the filter is¹²

$$\frac{1}{1 + \gamma\sigma_{pq}^2/|\mathcal{V}_{pq}|^2} \quad (3.14)$$

The parameter γ is not strictly a free parameter, in that its value is fixed by the equality constraint in the minimization. But as that norm difference was set to an arbitrary constant in the first place, we can view this one as the free parameter and the other as the derived one. Even if this were not so, γ is often treated as a free parameter in spite of the formal constraint, and it normally serves as a means of controlling the relative importance between the snr maximization and the more usual noise constraint.

Wiener filtering normally operates on a single image. One makes an estimate of the spectral signal and noise and possibly assumes a linear blur to be inverted so far as possible, and then the components of that image are reweighted to achieve the desired minimization. Here, we determine the filter coefficients by looking at a completely different image than the dirty map, (a point source), and apply them to the dirty map. In filtering parlance, we are using a nontypical prototype image. Alternatively, we can claim that it is the dirty beam we are filtering, and that the dirty map is being filtering in the same way to maintain the convolution relationship. In this case, the true “signal power” really is unity. By either justification, setting $\mathcal{V}_{pq} \equiv 1$ in equation 3.14 results in an expression for the filter exactly equivalent to robust weighting.

It is not clear exactly what the minimization means when robust weighting is derived via this formalism. Constraining the norm difference between the naturally weighted and robust weighted map to a constant involves both the sidelobes of the beam and the noise. It is also functioning to keep the weight sum nonzero, though the weight sum is not strictly held constant. That the functional form of robust weighting can be recovered clearly indicates a connection between robust and the parametric Wiener filter, but I do not believe the formalisms are strictly equivalent. At the least, there seems little physical motivation for derivation of robustness in this way and certainly the filter formalism is being used in a very nonstandard way. More work could certainly be done in connecting the theories of interferometric weighting and image restoration.

¹²We make the identification of the noise power, $P_n(w_x, w_y)$, with the variance of the gridded visibilities. The signal power is clearly the modulus of the true sky visibility at the grid points, and we do not care what the filter is at grid points where there are no sampled visibilities. The original filter to be inverted, H , is a delta function, and its transform in Fourier space is unity. In the terminology from Andrews & Hunt (1977), the objective function is $W(I^R) = \|\phi_r^{-1/2}\phi_n^{1/2}I^R\|^2 + \lambda(\|I^N - I^R\|^2 - const)$, where I^R is the robust weighted dirty map and ϕ_r is its covariance matrix. ϕ_n is the covariance matrix of the noise and I^N is the naturally weighted dirty map.

3.6.2 Specification of the Robustness

Robust weighting can be specified directly in terms of the strength of the fictitious point source S in real flux density units, along with the conversion constant from snr weights to flux density, ΔS_{uw} . This is a self consistent and passably intuitive way to specify the robustness, in that the minimization problem that is being solved is made manifest. It is also decidedly inconvenient, in that there is little indication of just how high the point source should be in order to achieve a given degree of robustness or beam shaping.

A second approach is to abandon the flux scale interpretation of S and instead specify it in terms of the gridded weights. With $S = (2/w_{targ})^{1/2} \Delta S_{uw}$, visibilities with $\sum_k w_k = w_{targ}$ will be downweighted by 1/2 compared to uniform weighting. The SDE task `uvmap` can write out an image of the gridded weights, prior to grid correction. When used with a boxcar convolution function, this is a very powerful means of examining the effect of a given weighting on the transfer function. Specifying S in terms of the gridded weights is really only useful with such an image as a guide, but it can provide very detailed control over robustness used to control the striping instability. The weight image is useful in general to gain insight on how the weighting is affecting the image.

For the more usual problem of achieving a reasonable tradeoff of resolution against sensitivity, it is useful to have a normalized form of the parameter. We don't really need to work in terms of S at all. What we really want is a knob that can be turned, with a convenient and known range where most of the action happens. Since the robustness changes its behavior based on σ_{pq} , it would seem reasonable to specify S proportionally to some average $\overline{w_{pq}}$. We don't need an absolute flux scale, so instead we choose to work with some average value of the gridded snr weights. Experimentally, S varied by about 4 orders of magnitude as weighting went from the limiting cases of natural to uniform weighting. For convenience in specification, we define the normalized robustness parameter to be logarithmic. For the zero point normalization, the value of S was found for all examined cases that produced an RMS thermal noise exactly halfway between that of the limiting cases. The scaling factor between the normalized S and this value was found to be reasonably constant and a convenient value was chosen and written into the definition of the normalized robustness. The final result is a normalized robustness parameter, R , which produces something close to natural weighting (most robust against striping) at a value of $R = +2$, something close to uniform weighting at $R = -2$, and some intermediate weighting at $R = 0$. Any individual case will have to determined by experimentation, but this gives the astronomer a reasonable range to examine. The definition of R is

$$S^2 = (\overline{w_{pq}})^{-1} (5 \cdot 10^{-R})^2 \quad (3.15)$$

where

$$\overline{w_{pq}} = \frac{1}{N_{vis}} \sum_{j=1}^{N_{vis}} \sum_{k \in c_{pq}} w_k \quad (3.16)$$

$\overline{w_{pq}}$ is the average snr weight sum over visibilities, not over gridding cells. Several forms of “average weight” were considered, and this one selected as being most resistant to fluctuations due to isolated points. Unfortunately, it is also slightly less convenient to evaluate than other choices. When used with the gridless weighting algorithm, this expression (where the inner sum is over the elliptical centered neighborhood) can be conveniently evaluated directly. For the gridded algorithm, it is the sum of the snr weights in each cell which is available after the first gridding pass. An additional pass through the visibilities will allow equation 3.16 to be used, but this was not deemed necessary. The **SDE** gridded routines actually use the simplified form

$$\overline{w_{pq}} \approx \sum_{p,q} \left(\sum_{k \in c_{pq}} w_k \right)^2 / \sum_{j=1}^{N_{vis}} w_j, \quad (3.17)$$

which is derived from equation 3.16 via the additional approximation that all snr weights are equal. Either form should suffice in new implementations, though which ever choice is used should be well documented.

In exactly the same way as super-uniform weighting is uniform weighting minimized over a smaller region in the image plane, we can extend robust weighting by minimizing over different regions. As before, this leads to a larger gridding box when determining the density weights via equation 3.13, (or a larger centered ellipse when using the analogous gridless equation).

We can now look at some examples of the basic tradeoff, again using the AT Full Track data. Many more examples are given in Section 3.7. The simplest display method is just to plot the various quantities against the robustness, overlaying two traces to show the tradeoff. Such a plot is given in Figure 3.6. The primary advantage to this style is that it shows the eccentricity of the beam very nicely. With asymmetric $u-v$ coverages, a common feature is that uniformly weighted beams are more circular than naturally weighted beams. To some extent this can be controlled with tapering as shown in Section 3.11, but with sufficiently poor coverage some degree of eccentricity is unavoidable. In general, the naturally weighted beam will be more eccentric than the uniform beam, and this sort of display makes the smooth increase in eccentricity quite manifest. It also shows quite graphically how the two parameters are anticorrelated. Unfortunately, it is inconvenient to use for robustness selection.

The primary parameters of interest here are the normalized thermal noise and the mean fitted beamwidth. The robustness itself is unimportant except that it allows us to select positions along the curve. Hence all further plots of this tradeoff are of the two primary quantities against each other, parametrically indexed by the robustness. Such a plot of the same data is given in Figure 3.7. Once again we have recovered a curve of the same general shape as that of Figure 3.1.

The useful regimes in robust weighting are usually obtained by moving inwards from the endpoints on the tradeoff curve. Moving in from the uniform weighting limit,

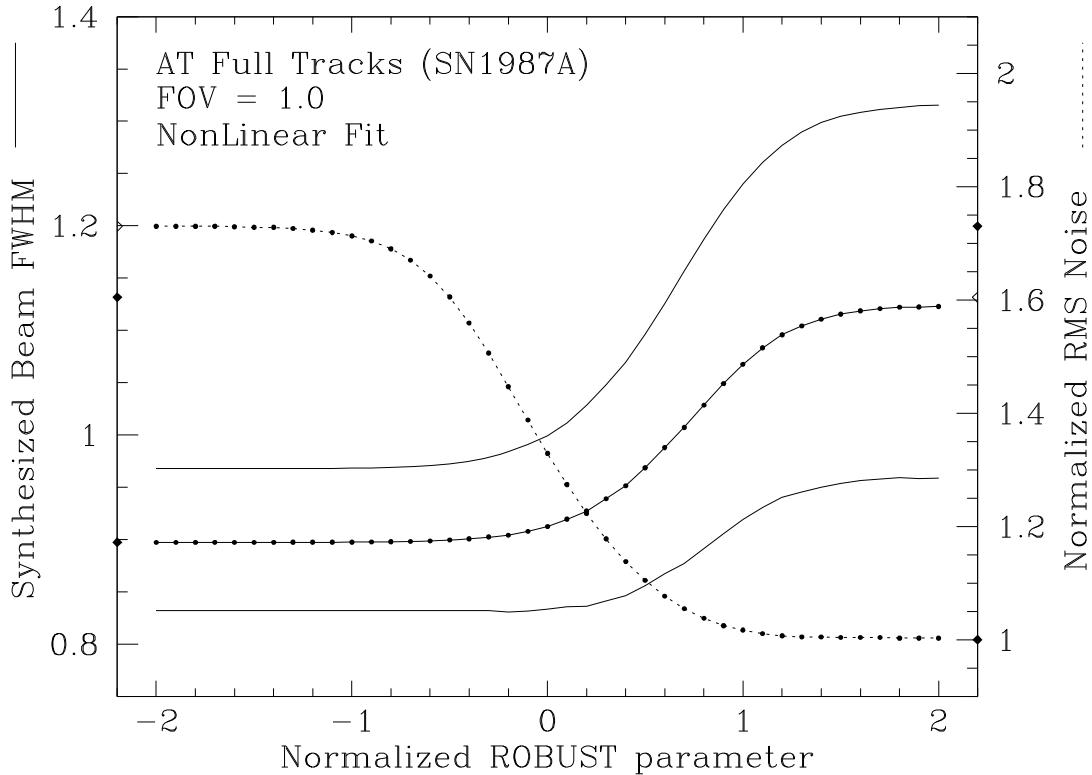


Figure 3.6: Thermal Noise-Resolution robustness tradeoff. The three traces for beam FWHM are B_{maj} , B_{min} and the geometric mean. This style of plot shows the beam eccentricity very nicely, but is difficult to use for practical robustness selection.

one can often obtain a significant decrease in thermal RMS while giving up very little in resolution. In Figure 3.7, this might be a robustness of 0. Working downwards from the naturally weighted limit, a robustness of 1 results in a somewhat improved resolution with almost no less of sensitivity. We might also choose to work in the regime near 0.5, with intermediate properties in both parameters. This compromise beam may be particularly useful, as it often has the most desirable sidelobe properties of any of the weightings. Nearly all cases with good $u-v$ coverage have these basic regimes.

3.6.3 Example Thermal Noise Calculations

We can gain a more intuitive feeling for why these regimes appear by considering some simple cases. We will qualitatively explain the weight data from the VLA Full Track case of Figure 3.12a. In figures 3.8 through 3.10 we show the azimuthally averaged total gridded weights, the azimuthally averaged total density weights, and the histogram of the ungridded density weights. The azimuthal averaging was done as a simple mean over all the pixels which fall into each radial bin. The total weights are averaged over both the weighting bins with nonzero occupancy and all of the cells. (The former shows the form of

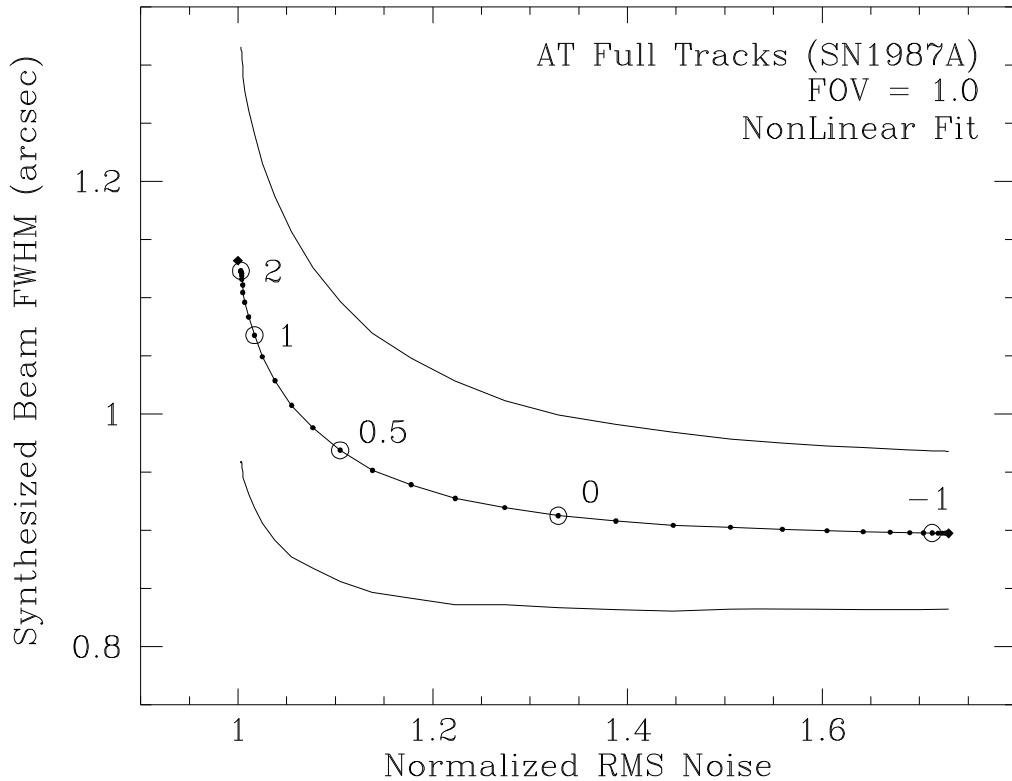


Figure 3.7: This is a better way to display the same data as in Figure 3.6, but the beam eccentricity is hard to read and suppressed in later plots. Circled values are produced by the indicated value of the normalized robustness. Note that the vertical axis is rather compressed. The curve is actually quite symmetric. If the range is autoscaled and only the mean beam is plotted against thermal RMS, the shape of the curve is almost unchanged after reversing the axes.

the weighting applied to the data, the latter shows the approximate transform of the beam more clearly.) The density weights are azimuthally averaged over occupied cells only.

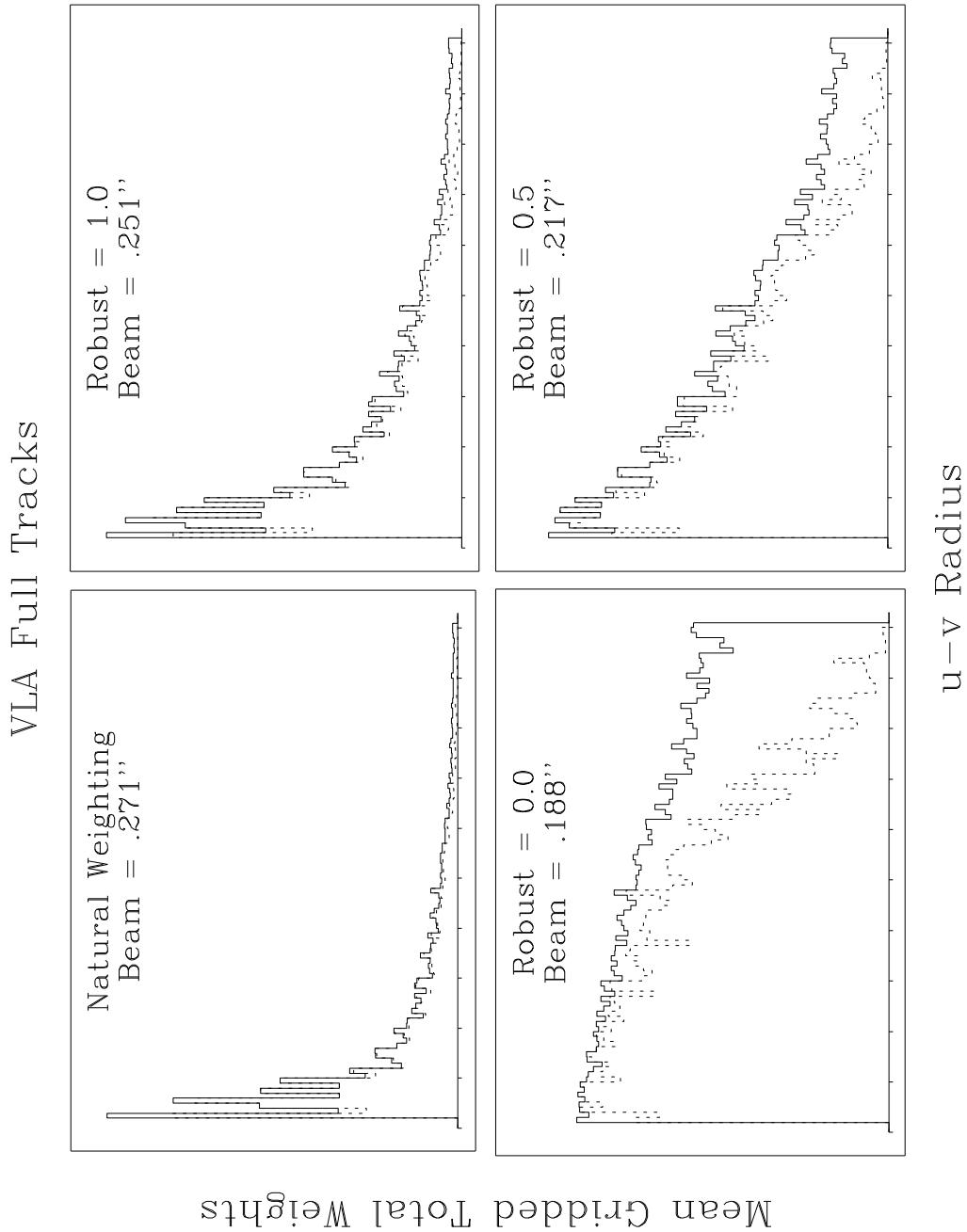


Figure 3.8: Azimuthally Averaged Gridded Weights — VLA Full Tracks. Uniform weighting would be perfectly flat on this diagram and has a beamwidth of $.181''$. There are 10 bins per tick mark, and the lowest bin begins at zero. The solid trace is averaged over positive weighting cells only. The dotted trace is averaged over all cells.

VLA Full Tracks

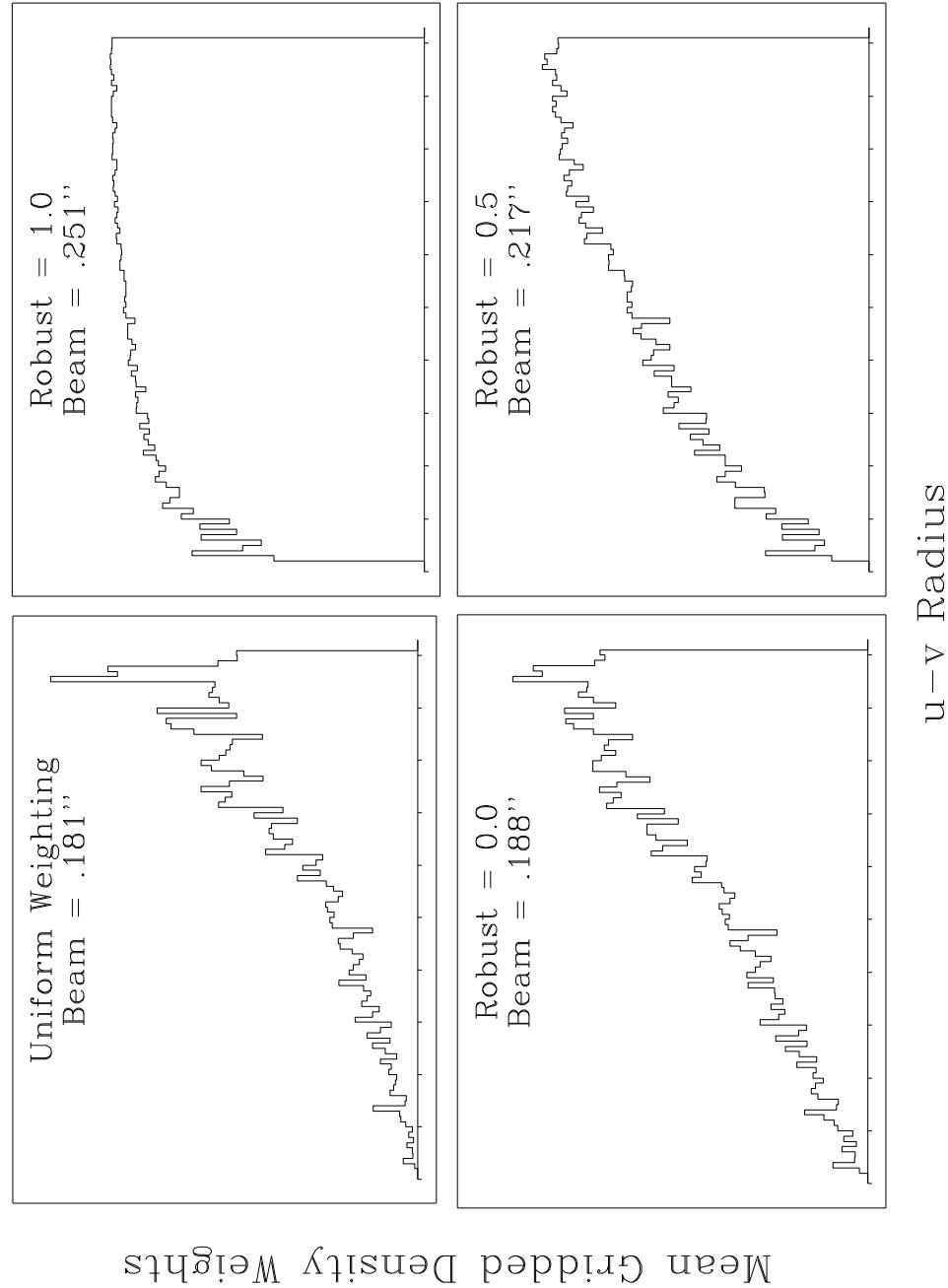


Figure 3.9: Azimuthally Averaged Gridded Density Weights — VLA Full Tracks. Note that uniform weighting is shown in the same panel as natural weighting in the previous figure. Natural weighting would be perfectly flat on this diagram. The natural beamwidth is $.271''$. There are 10 bins per tick mark, and the lowest bin begins at zero.

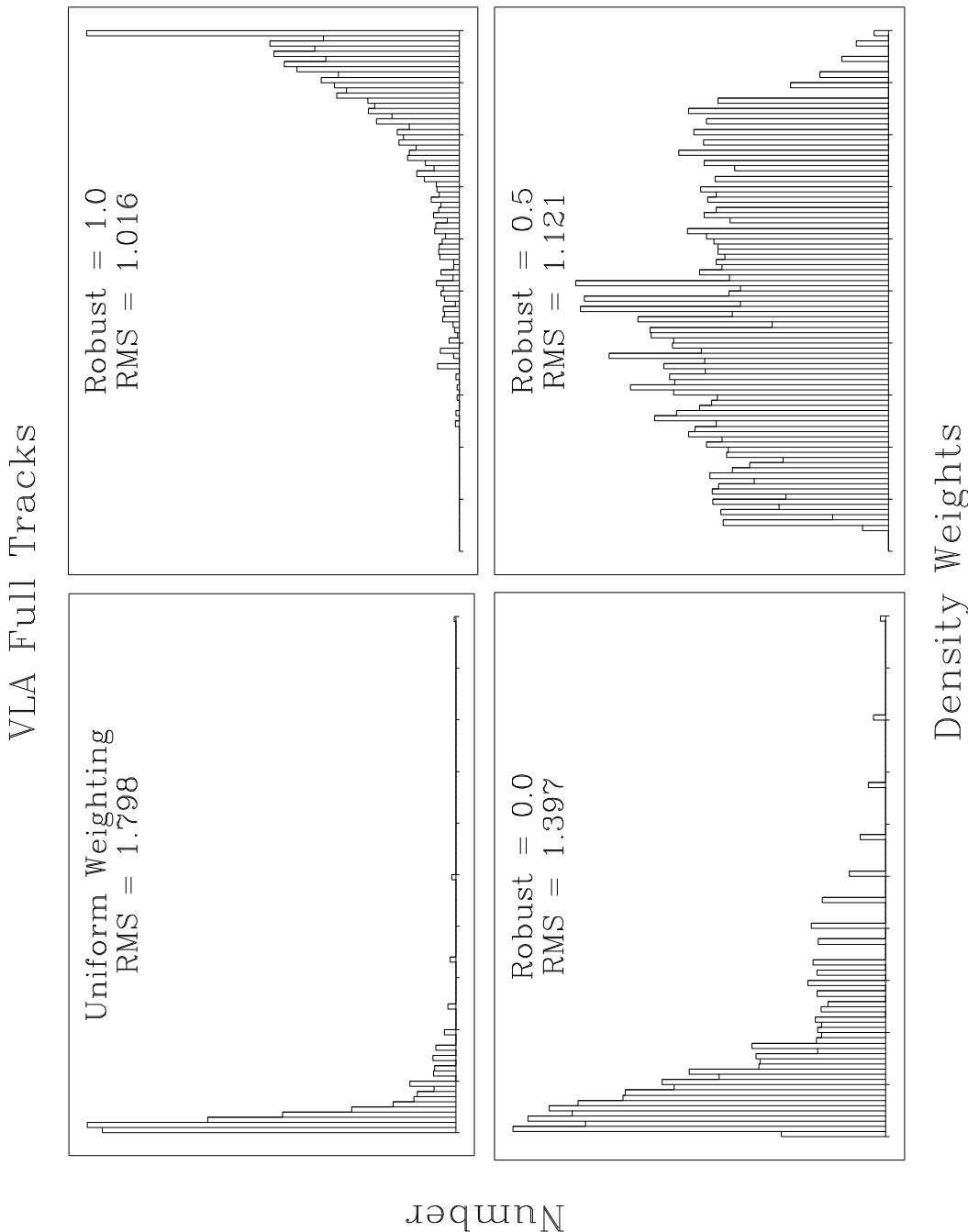


Figure 3.10: Density Weight Histograms — VLA Full Tracks. The naturally weighted histogram is not shown and would be a delta function. The naturally weighted normalized RMS is 1.00 by definition. There are 10 bins per tick mark, and the lowest bin begins at zero.

It is not completely intuitive why these plots look as they do. Consider three gridded snr weights such that $w_0 \gg w_1 \gg w_2$. These will likely lie at increasing distance from the u - v plane origin, but not necessarily. From equation 3.13 the ratio of any two density weights at a given robustness is given by

$$\frac{D_i}{D_j} = \frac{S^2 w_j + 2\Delta S_{uw}^2}{S^2 w_i + 2\Delta S_{uw}^2},$$

where S is the robustness parameter specified in absolute flux units. Let us start from the naturally weighted limit and examine the ratios D_1/D_0 and D_2/D_0 . As S increases from 0, the term $S^2 w_0$ will become significant compared to $2\Delta S_{uw}^2$ before either $S^2 w_1$ or $S^2 w_2$. At that value of S , we have the relation

$$D_1/D_0 \approx D_2/D_0 \approx 1 + \frac{w_0 S^2}{2\Delta S_{uw}^2} > 1$$

D_1 and D_2 have both increased relative to D_0 , and by the same factor. Since the weighting is invariant to a constant rescaling, we can divide all weights by the above quantity and see that the effect of the robustness is to drop the density weight D_0 relative to D_1 and D_2 which remain fixed. Moving in from the natural limit, the effect of robustness is to drop the total weights at the location of the largest gridded snr weights. A similar argument can be made for the ratios D_2/D_0 and D_2/D_1 as S decreases from infinity. The effect of robustness moving in from the uniform limit is to drop the total weights at the location of the smallest gridded snr weights.

There is one important difference in the behavior of these two limits. The VLA and nearly all arrays that have not been specifically designed to avoid it have an overabundance¹³ of short spacings. The shortest spacings present will almost certainly have the highest gridded snr weights. This is true even for the highly unbalanced array cases of Figure 3.12b, where the sensitivity of different array elements varies greatly. In the natural weighting limit, the total weights downweighted by the robustness are consequently quite localized near the origin of the u - v plane. It is exactly the overabundance of these low spatial frequencies which are responsible for the wide naturally weighted beamwidth, and downweighting them relative to the higher frequencies causes the beamwidth to vary rapidly with robustness. The Robust=1.0 and Robust=0.5 panels in Figure 3.8 show how little one has to modify the central spike in the weight distribution in order to achieve a significant change in the beamwidth.

In contrast, the u - v locations of particularly low gridded snr weights are not localized. There is a tendency for these low points to appear towards the outer regions of the u - v plane, but this is only in a probabilistic sense. The locus all of all points with an occupancy of one is scattered evenly and sparsely throughout the sampling, with a smooth

¹³The very shortest spacings may be missing, but that is of no consequence here.

gradient increasing towards the higher spatial frequencies. Points with an occupancy of two are similarly distributed, except that the gradient is less steep, and so on for higher occupancies. The density weights derived from the gridded snr weights are remarkably regular. The averaged density weights in Figure 3.9 are quite typical, even for the unbalanced arrays. The resulting total weight modification after a mild increase in robustness from the uniform limit is a smooth and gentle roll off in the higher frequencies. While partly due to the low occupancy of the outer regions,¹⁴ this smooth downweighting does not affect the beamwidth particularly significantly.

Finally, consider the density weight histograms in Figure 3.10. For the analytic examples, we treat only the case where all the snr weights are equal and without taper. Hence the normalized RMS of equation 3.6 becomes a function solely of the density weights.

$$RMS = \sqrt{N \sum_k D_k^2} / \sum_j D_j \quad (3.18)$$

In Figure 3.11, we have three analytically tractable density weight distributions, roughly representative of the regimes encountered in robust weighting.

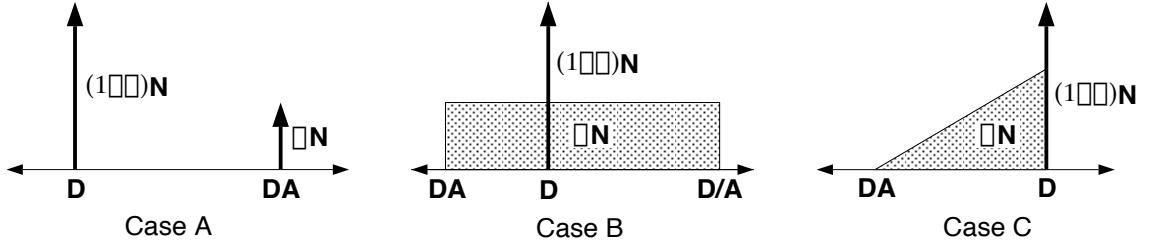


Figure 3.11: Model density weight histograms.

In case A, we merely have the bulk of the weights concentrated at some density weight D , and a small fraction η of the weights which have been amplified by a factor A . Very crudely, this is similar to what happens in uniform weighting.

For this case we have

$$\begin{aligned} \sum_j D_j &= (1 - \eta)ND + \eta NAD \\ \sum_k D_k^2 &= (1 - \eta)ND^2 + \eta N(AD)^2 \\ RMS &= \sqrt{1 - \eta + \eta A^2} / (1 - \eta + \eta A) \end{aligned}$$

¹⁴There are less visibilities in the outer u - v plane, so a comparatively drastic change in the weights there can produce less effect on the beam than a milder change in a more heavily occupied area.

Inserting some representative values, we obtain the following matrix. The last column is particularly interesting, since this is the value of A for which the RMS is equal to that produced by simply dropping the amplified points all together. From a signal-to-noise perspective, is it actually *worse* to add in additional data that are badly amplified beyond their snr weights, than it is to ignore it.

$\eta \backslash A$.2	.5	1	2	5	10	20	40	$A(RMS = \sqrt{1/(1-\eta)})$
.5	1.20	1.05	1.00	1.05	1.20	1.29	1.34	1.38	∞
.3	1.11	1.04	1.00	1.06	1.30	1.50	1.64	1.73	3.5
.1	1.03	1.01	1.00	1.04	1.32	1.74	2.21	2.59	2.25
.01	1.00	1.00	1.00	1.00	1.07	1.29	1.88	2.97	2.02
.001	1.00	1.00	1.00	1.00	1.00	1.04	1.16	1.55	2.00
.0001	1.00	1.00	1.00	1.00	1.00	1.00	1.02	1.07	2.00

Table 3.4: Analytical RMS for case A

Reading off some representative numbers from Figure 3.10, we see that the ratio between the mean weight and the highest weight is something like 30 for uniform weighting and 10 for a robustness of 0. The RMS from the actual simulation is 1.798 and 1.397 respectively. There is a tail of high weight points, rather than a single high weight, but assigning a collective weight of a percent or somewhat less seems reasonable. In the toy analytic distribution, note that 1% of the data amplified by a factor of 20 beyond the rest produces a degradation in the RMS noise of 1.88. Amplified by a factor of 10, the RMS drops to 1.29. The match to the actual numbers is coincidentally rather good, but clearly the high end tail of the density weights is *very* important to the RMS. The combination of the strong dependence of the highly amplified weights on the robustness, (and hence the RMS on the robustness), along with the gentle dependence of the beamwidth on robustness yields the characteristic uniform weighting limit of the robustness tradeoff curves.

Case B is the rectangular distribution of density weights with a central condensation. Obviously, we are working in the large N limit, where the weight histogram can be approximated by a continuous distribution. Let $n(t) dt$ be the number of weights between t and $t + dt$. For this problem, we take $n(t)$ to be constant on the interval $(DA, D/A)$, and find the magnitude n from the requirement that the integral of $n(t)$ over this integral be equal to ηN . In this approximation,

$$\int_{DA}^{D/A} n(t) dt = n(D/A - DA) = \eta N$$

$$\begin{aligned} \sum_j D_j &= (1 - \eta)ND + \int_{DA}^{D/A} t n(t) dt \\ &= ND(1 - \eta + \frac{\eta}{2(A - 1/A)}(1/A^2 - A^2)) \end{aligned}$$

$$\begin{aligned}
\sum_k D_k^2 &= (1 - \eta)ND^2 + \int_{DA}^{D/A} t^2 n(t) dt \\
&= ND^2(1 - \eta + \frac{\eta}{3(A - 1/A)}(1/A^3 - A^3)) \\
RMS &= \sqrt{1 - \eta + \eta \frac{(1/A^3 - A^3)}{3(1/A - A)}} / \left(1 - \eta + \eta \frac{(1/A^2 - A^2)}{2(1/A - A)}\right)
\end{aligned} \tag{3.19}$$

$\eta \diagup A$	1	1.5	2	4	10	20	40
1.0	1.00	1.02	1.06	1.12	1.15	1.15	1.15
0.5	1.00	1.01	1.04	1.17	1.38	1.49	1.56
0.3	1.00	1.01	1.03	1.16	1.48	1.72	1.89
0.1	1.00	1.00	1.01	1.09	1.47	1.99	2.54
0.01	1.00	1.00	1.00	1.01	1.11	1.40	2.11

Table 3.5: Analytical RMS for case B

This case is obviously intended to represent the intermediate robustness distribution, which is characteristically rather flat. Estimating that the ratio of minimum to maximum weights in the appropriate panel of Figure 3.10 is about $90/5 \approx 4^2$, and that there is not really a significant condensation, we predict an RMS of 1.12 which is embarrassingly close to the actual RMS of 1.121.

Case C is the triangular distribution of weights, with the concentration off to one side.

$$n(t) \equiv at + b \quad \& \quad n(AD) \equiv 0 \quad \Rightarrow \quad b = -aAD$$

$$\int_{DA}^D n(t) dt = n(D/A - DA) = \eta N \quad \Rightarrow \quad a = \frac{2\eta N}{D^2(A - 1)^2}$$

$$\begin{aligned}
\sum_j D_j &= ND(1 - \eta) + \frac{1}{3}aD^3(1 - A^3) + \frac{1}{2}(-aAD)D^2A(1 - A^2) \\
&= ND(1 - \eta) + aD^3\frac{1}{6}(A + 2)(A - 1)^2 \\
&= ND(1 - \eta) + \frac{1}{3}\eta ND(A + 2) = ND(1 - \frac{1}{3}\eta + \frac{1}{3}\eta A)
\end{aligned}$$

$$\begin{aligned}
\sum_k D_k^2 &= ND^2(1 - \eta) + \frac{1}{4}aD^4(1 - A^4) + \frac{1}{3}(-aAD)D^3(1 - A^3) \\
&= ND^2(1 - \eta) + \eta ND^2\frac{1}{6}(A^2 + 2A + 3) \\
&= ND^2(1 - \frac{1}{2}\eta + \frac{1}{3}\eta A + \frac{1}{6}\eta A^2)
\end{aligned}$$

$$RMS = \sqrt{1 - \frac{1}{2}\eta + \frac{1}{3}\eta A + \frac{1}{6}\eta A^2} / \left(1 - \frac{1}{3}\eta + \frac{1}{3}\eta A\right) \tag{3.20}$$

$\eta \backslash A$	0	0.1	0.2	0.5	1	2	5	10	20	40
1.0	1.06	1.04	1.03	1.01	1.00	1.02	1.08	1.13	1.17	1.20
0.5	1.04	1.03	1.02	1.01	1.00	1.02	1.15	1.31	1.47	1.58
0.3	1.02	1.02	1.02	1.01	1.00	1.02	1.15	1.38	1.65	1.88
0.1	1.01	1.01	1.01	1.00	1.00	1.01	1.09	1.32	1.76	2.34
0.01	1.00	1.00	1.00	1.00	1.00	1.00	1.01	1.06	1.24	1.72

Table 3.6: Analytical RMS for case C

This distribution is roughly similar to the high robustness case, though perhaps a more extended tail would have been better. Reading off from the plot again, values of $A = .5$ and η approaching 1 seem reasonable. Given the crude model, about all we can say here is that this predicts an RMS very close to 1. The actual measured value is 1.016. The very mild dependence of RMS on robustness in this limit, along with the strong downweighting of low spatial frequencies and consequent dependence of beamwidth on robustness, is responsible for the behavior of the robustness RMS/resolution tradeoff curve in the natural weighting limit.

3.7 Simulations

We now address the more pragmatic question of “what can robustness do for *my* data?” A number of different array geometries were simulated, and the basic tradeoff of resolution against point source sensitivity mapped out by varying the robustness. An attempt was made to select a reasonable number of geometries that roughly span the typical parameter space encountered in practice, but of course any given case should be examined on its own particulars. Highly degenerate coverages produced by low declination sources were not well studied, but in additional simulations varying the imaging parameters for the geometries given here, there were no significant surprises. These results may be taken as reasonably typical for nonpathological samplings.

The geometries include several variations on VLA coverages: full track, snapshot and multi-configuration. A two track observation from the Australia Telescope Compact Array shows some properties of east-west arrays, as well as the effects of high oversampling. Both simulated ‘perfect’ observations and data from real observations are given here; real observations have names with the observation target appended in parentheses. A full track VLBA observation is presented. Two unbalanced VLBI arrays were included, as this may well be where robust weighting is most useful. The VLBA plus the phased VLA plus the Green Bank Telescope is an example of an array of modest sized elements with the sensitivity completely dominated by a single baseline. Finally, this same array with an additional orbital element in a VSOP-like orbit is an example of a case where the bulk of the u - v coverage is composed of very low sensitivity baselines. In these unbalanced cases,

the sensitivities were simply taken to be a function of the telescope effective diameter.¹⁵ No attempt was made to accurately simulate different receiver temperatures or the like. The VLA is described in Thompson et al. (1980), the AT in Frater et al. (1992), and the VLBA in Napier et al. (1994). The only Green Bank Telescope reference currently extant is the original “Request for Proposals” document, (National Radio Astronomy Observatory, 1990). The spacecraft VSOP is described in Hirabayashi (1990).

Several of the coverages were taken from projects in progress, and the imaging parameters somewhat reflect my current work with super resolution and high fidelity mapping — the relatively small image sizes and high oversampling in the SN1987A and 3C48 cases mean that the size of the $u-v$ gridding cells (which governs the effective degree of super-uniform weighting) is somewhat higher than might be encountered by people studying large diffuse emission. In an effort to be somewhat more representative, the synthetic VLA cases were done with a lower degree of oversampling. The orbiter is a somewhat pathological case, and the tapering techniques described later were found to converge better with the fairly high degree of oversampling used here.

It is certainly possible to present other beam parameters such as most negative sidelobe, or highest off-peak sidelobe against robustness, but these were often found to be somewhat misleading in tabular or graphical form. For instance, in general one wishes the most negative sidelobe to be small in absolute magnitude. But by this criteria, the ‘best’ beam is usually the naturally weighted beam. Unfortunately, the natural beam usually gains this property by sitting the entire inner region of the beam on an extended shelf, which masks the region where the worst sidelobes are to be found. Unfortunately for the deconvolution, this property of the beam can be worse than the negative sidelobes. Since addressing the best beam shape is still a somewhat subjective process, we present here slices through the beam at the position angle of the fitted¹⁶ major and minor axes. The exact position angle selected by the fit is sometimes a bit unfortunate. Arguably a better method would be to select “interesting” position angles by hand, but this is quite subjective, difficult to automate and still more difficult to describe. For better or worse, these plots use the fitted position angle, and the actual slices taken are shown on the corresponding contour plots as the light dotted lines. The sole exceptions to this are the snapshot plots, since in this case there is a well defined position angle. These slices are aligned so that one slice is down the sidelobe nearest to east-west. The slices are indexed by normalized robustness, and include the two limiting cases of uniform and natural weighting. The numbers found to the right of the plots are the normalized thermal RMS of equation 3.6. Visibility data sets were taken from actual observations or produced by the SDE task `vissim` or the Caltech

¹⁵See equations 4 and 11 of Walker (1989) for details. Briefly, the antenna sensitivity K_i is proportional to effective antenna area or D_i^2 . The thermal noise variance is proportional to the inverse sensitivities, $1/K_1 K_2$. Thus $w_k \propto 1/\sigma_k^2 \propto D_1^2 D_2^2$.

¹⁶The fits are done with the nonlinear fitting algorithm described in Appendix B.

VLBI package program `fake`. The sets from `fake` were then reweighted with the SDE task `visrewt`. All beams were produced by the SDE task `uvmap`. Except where noted otherwise, these are all done with gridded weighting and the normalization of the robustness done with equations 3.15 and 3.16. The textual descriptions and discussion of each simulation are collected first, and these are followed by plots of the beams at different robustness in figures 3.13 through 3.22.

Name	N_{vis}	Δt (s)	w_{min}	w_{avg}	w_{max}	N_{pix}	$\Delta\ell$ ($''$)	over- samp	δ ($^{\circ}$)
VLA Full	335907	30	1	1	1	512	.05	2.08	35
VLA 3C48	108491	60	22.9	132.5	160.7	512	.03	3.68	33.2
VLA Snap 30	10530	10	1	1	1	512	.05	2.20	35
VLA Snap 5	1755	60	1	1	1	512	.05	2.20	35
AT Full SN87A	116396	15	1	1.34	2	512	.1	6.02	-69.3
VLBA Full	93634	30	1	1	1	512	.001	2.42	50
+ VLA/GBT	49084	60	39.06	449.1	13225	512	.001	2.54	40
+ Orbiter	59397	60	6.25	375.3	13225	512	.0002	4.26	40
VLA Multi M82	1369989	—	.218	.639	1.00	1024	.02	2.10	69.9
B config	330875	10	1.00	1.00	1.00	(May 1993)			
A config	202959	10	.585	.585	.585	(Dec 1992)			
BnA config	485883	10	.621	.621	.621	(Dec 1991)			
B config	178490	10	.301	.301	.301	(Dec 1987)			
D config	59080	30	.854	.854	.854	(Dec 1985)			
C config	112702	30	.218	.218	.218	(Sep 1985)			

Table 3.7: Simulation Parameters

3.7.1 VLA Full Tracks

Synthetic data produced by SDE `vissim`, VLA A configuration, 8.4 GHz, 35° declination, 30 second integrations, 8 hours of data, excellent u-v coverage, moderate oversampling

With a robustness of 0, one can go from an RMS of 1.80 to 1.40, while broadening the beam only slightly from $.181''$ to $.187''$. This corresponds to an effective increase in observing time of $(1.8/1.4)^2 - 1 = 65\%$. The strong negative sidelobes are softened slightly, essentially without penalty in positive sidelobes. (Note the position angle of the slices.) The gains are modest, but there is no reason not to use this beam in preference to a uniformly weighted one. If one increases the robustness further to 0.5, one obtains the compromise beam of 1.12 RMS and $.217''$ beamwidth. The positive sidelobes have increased only slightly, the negatives have nearly disappeared. This beam has an excellent match to its own fitted Gaussian, providing a high fidelity flux scale for weak sources. This is an excellent general purpose beam, providing increased resolution, better sidelobes, and almost the sensitivity of

the natural beam. (See the discussion about automatic robustness selection in Section 3.9.) The naturally weighted beamwidth is $.271''$. This sampling is in the regime where super-uniform weighting or image size makes a considerable difference in the thermal RMS. The uniformly weighted thermal RMS with a weighting field of $1/4$ is 2.72 , with a corresponding beamwidth of $.159''$. If the weighting field of view is increased to 4 , (or the image size is increased to 2048^2), the thermal RMS/resolution decreases to $1.37/.198''$. The practical effects of super uniform weighting is examined in Section 3.8.4.

3.7.2 VLA Full Tracks (3C48)

Observational data of 3C48 courtesy of R. Craig Walker. VLA A configuration, 8.4 GHz, 33.2° declination, 60 second integrations, 5.5 hours of data spanning 13 hours on 22 August 91 for a duty cycle of $\sim 45\%$, excellent u-v coverage, high oversampling

This is the data used in Chapter 7, and is a “real” observation that is otherwise very similar to the previous case. With all the starts and stops for calibration and frequency changes, this allows more places for visibility points to be isolated than in the previous synthetic set, and also gives a patchier coverage of the outer spacings. The higher degree of oversampling in the image plane will tend to increase the noise, and the time averaging down a factor of two in time will tend to decrease the noise. (See Section 3.8.3 for more discussion of time averaging.) The results are qualitatively very similar to before and by chance nearly quantitatively so. We have the same regimes, with a modest decrease in the RMS at nearly no impact on the uniformly weighted beam, and a compromise beam with degraded resolution but nearly natural thermal noise and excellent sidelobes. Since this sampling is somewhat more asymmetric than the perfect set, the beam is more eccentric as well. Notice that the compromise beam is less perfect, since the rising negative sidelobes along the major and minor axis do not pass through zero at the same value of robustness. When the two data sets are rigorously compared by averaging the synthetic set down in time and imaging both on this grid, we find that indeed the real coverage has taken a slight performance degradation from the imperfect sampling. The fitted beamwidth is larger due to lower weighting of the outer spacing after uniform weighting, ($.194''$ compared to $.171''$ for the synthetic set), and yet the RMS noise is worse. Uniform RMS is 1.97 for this set, and 1.91 for the synthetic set.

3.7.3 VLA 30 Point Snapshot

Synthetic data produced by SDE vissim, 10 second integrations, VLA A configuration, 8.4 GHz, 35° declination, 5 minutes of data at zenith, snapshot u-v coverage, moderate oversampling

This simulation is included to demonstrate that even with rather sparse u - v coverage, robust weighting can still provide significant sensitivity improvement. Robustness for snapshots is most effective when there are a significant number of points in each track,

and this is a rather extreme example with 10 second averages. The same regimes mentioned in the full track cases are still present: the uniformly weighted RMS is 1.53, and can be dropped to 1.30 with little change in beam shape. (The mean beamwidth goes from $.216''$ to $.220''$.) There is a beam with an RMS/resolution of $1.07/.244''$ with very slightly improved resolution and better beam shape than the natural beam with $1.00/.265''$, though the wings of the beam are already somewhat higher than desirable. For snapshots, other parameters are as important as robustness and appropriate combinations of conventional parameters can produce nearly comparable sensitivities. The very best beams will still use a degree of robustness in addition to the usual parameters.

Notice that for nearly all snapshot coverages and imaging parameters, the u - v weighting box size (8000λ in this example) is much smaller than the typical distance between clusters in the u - v plane. Most points gridded together are from the same baseline. There is little interaction between different clusters. Consequently the weighting will be dominated by the details of how the points happen to fall on the uniform weighting grid. Since every point is a member of a single isolated short track, the snapshot coverage is particularly vulnerable to gridding irregularities. In the case studied, the longest spacing will sweep over just 2.7 gridding cells in the 5 minutes of observation. Shorter spacings may fall onto only one or two cells in the course of a track. Super-uniform weighting, which controls the size of the gridding cell, is of major importance for snapshots.

It has previously been appreciated, (Sramek and Schwab, 1989, p.123 and Figure 6-3), that super-uniform weighting often leads to favorable beam properties for VLA snapshots. One is tempted to believe that it is somehow “smoothing over” the large gaps in coverage. Yet with moderate increases in the gridding cell size, one is actually smoothing over small scale gridding irregularities rather than gross features of the sampling. Still, because of the particular vulnerability of snapshot to gridding, this result actually dominates over the more usual interpretation of super-uniform weighting as effectively upweighting the outer spacings. The normalized RMS can actually go down with a higher degree of super-uniform weighting, in contrast to the typical behavior of a dense sampling. If we decrease the weighting field of view to 1/4 for this sampling, (increased super-uniform weighting), the RMS drops from 1.53 to 1.46, while the mean beamwidth also drops from $.216''$ to $.209''$! The beam has also lost some of the shelf along the arms of the pattern. The only disadvantage is that the worst negative of the beam has gone from -8.1% to -9.2% . While the difference in beam parameters is slight, in almost all respects the super-uniformly weighted beam is superior to the uniform beam for the 30 point snapshot.

Super-uniform weighting is a good idea for snapshots.

It should be remembered that **SDE** uses the scaled cell method of super-uniform weighting. **AIPS** tasks which use approximately centered integer boxes should perform¹⁷

¹⁷The software to properly perform this comparison does not currently exist, since **AIPS** imaging tasks

somewhat better in terms of RMS. The gridless weighting scheme of the Section 3.5.1 is better still, since the RMS is so dominated by the gridding irregularities. A second set of beams from this data set is presented in Figure 3.16, using the gridless weighting and a weighting field of view of 1/4. The gridless weighting produces an improved RMS in nearly all cases, but when combined with a significant robustness the incremental improvement is usually slight. With snapshots, however, the improvement is dramatic. The super-uniform gridless beam has an RMS of only 1.11, and a beamwidth of .217''. The sidelobe pattern has the least shelf of any seen, and a maximum negative of only -6.9% . The gridless weighting has done a sufficiently good job here that there is little left for the robustness to improve. At a robustness of 0.5, the RMS/resolution becomes 1.06/.224'', but the extended shelf along the arms of the beam has gotten slightly worse. Either of these would be a reasonable choice depending on the specific needs of the project, but the robustness has not made a significant improvement over the beam provided by the gridless/super-uniform weighting alone.

Super-uniform weighting is an even better idea for snapshots when combined with gridless weighting.

3.7.4 VLA 5 Point Snapshot

Synthetic data produced by SDE viissim, 60 second integration time, VLA A configuration, 8.4 GHz, 35° declination, 5 minutes of data at zenith, snapshot u-v coverage, moderate oversampling, Weighting FOV = 1/4

The u - v coverage is not shown in Figure 3.12, but is visually equivalent to that of the 30 point snapshot. This simulation is completely equivalent to the previous one, after the data has been time averaged to one minute. In addition, the weighting field of view has again been reduced to 1/4.

Since there is little interaction between u - v tracks in a snapshot, and the visibilities in a given short track will be combined in the gridding stage anyway, there is little sensitivity lost by time averaging the visibilities prior to gridding. The only sub-optimal weighting will come at the slight overlap regions between the arms of the sampling pattern, and this is a minor effect. But time averaging ensures that no single integration can be isolated from the rest by the uniform weighting grid, which leads to more uniform weights across visibilities and better sensitivity.

Time averaging snap shots is a good idea, especially if care is taken to select an averaging time that divides the data into roughly equal sized time bins.

of 15JAN95 and earlier do not calculate the thermal RMS. SDE imaging tasks do not implement the integer box super-uniform weighting, since the need is largely obviated by gridless weighting.

Of course, one should try to ensure that single points are not isolated at the end of a scan by the time averaging itself. Even in such a case one would have to be doubly unlucky with a point isolated both by the time averaging and by gridding for a catastrophic reweighting to occur. Time averaging is explored further in Section 3.8.3.

The super-uniformly weighted beam is by itself a significant improvement over the uniform, $FOV_w = 1$ beam of the 30 point case, though not nearly as good as the gridless uniform beam. It has RMS/resolution of $1.25/.210''$ compared to the gridded uniform $1.53/.216''$, and the sidelobe structure is improved as well. Adding a robustness of 0.5, the RMS drops to 1.09 and the beamwidth increases to $.220''$. There is a very slight degradation of sidelobes as well. This beam is very similar to the gridless beam with robustness earlier. As robustness is added, the gridded and gridless weighting schemes approach each other.

Clearly one can go too far in time averaging a snapshot. If all 5 minutes of data are averaged to a single time point, there is essentially no difference at all between uniform and natural weighting at the normal $FOV_w = 1$ case. The uniform thermal RMS is 1.01, and the mean beamwidth $.217''$. The sidelobe shelf, of course, is awful. But when the weighting field of view is decreased again to 1/4, the beams look very much like those of the corresponding gridless weighting case, which are excellent. If one can tolerate the time smearing, (or if there is a very small amount of data to begin with), substantial time averaging can help snapshot coverage considerably. It is somewhat inflexible and there are better ways, but if the more exotic weightings are not available then time averaging and super-uniform weighting are considerably better than nothing.

Summarizing the snapshot results, the recommended approaches ranked roughly from best to worst are:

- Gridless + super-uniform weighting
(robustness and time averaging don't matter much)
- Super-uniform weighting + moderate time averaging + robustness
- Super-uniform weighting + heavy time averaging
- Super-uniform weighting
- Uniform or natural weighting

3.7.5 VLA Multiple Configurations (M82)

Observational data of M82 courtesy of Philip Kronberg, Richard Sramek and Michael Allen. 6 observations with the VLA spanning all configurations, 22.5 GHz, 69.9° declination. 30 second integrations on C & D configuration data and 10 second integrations otherwise. ~ 13.6 hours of data out of 36 hours spanned. Duty cycles range from $\sim 20\%$ to $\sim 58\%$ with an average of $\sim 38\%$. Superb coverage, moderate oversampling.

The weights for each individual data set do not reflect the integration times in a simple way, as the system temperatures dropped by more than a factor of two after an upgrade completed in 1989. In addition, K band system temperatures can be strongly affected by weather. Reliable effective system temperatures were not available for these data, so the relative weights were determined by Michael Allen and Michael Rupen in the following way. Remember that for the naturally weighed dirty image

$$\Delta I_{nat}^2 = \Delta S_{uw}^2 / \sum_j w_j,$$

where ΔS_{uw} is understood to be different for each data set. For the final weights w'_k to be statistical, we wish

$$\begin{aligned} w'_k &\propto 1 / \Delta S_k^2 \\ &\propto w_k / \Delta S_{uw}^2 \\ &\propto w_k / (\Delta I_{nat}^2 \sum_j w_j), \end{aligned}$$

with j indexing visibilities over a particular data set and k indexing visibilities over the composite data set. The ‘constant’ ΔS_{uw} above is understood to be a function of individual data set, and the proportionality constant for the combined weights is suppressed.

The modified weights were computed from a measured RMS in the dirty Stokes V naturally weighted images for each epoch and from the *AIPS* “sum of the gridded weights”. As with most sources, these Stokes V images are expected to be source free and provide a convenient measurement of the thermal noise. The modified weights were normalized by setting the weights of the most recent observation to unity. The image size has been increased to 1024^2 for these simulations, to more properly reflect typical uses for multi-configuration data sets. It may be somewhat smaller than typical even so.

This particular data set is actually something of a compromise, in that it is more heavily dominated by the larger configurations than many multiple configuration VLA observations. However, the very question of “what is the best ratio of observing time in different scaled array configurations?” is not easily answered. An obvious criterion would be to seek the most uniform coverage of the u - v plane possible. If the scaled array configurations do not overlap, the density of sampled u - v points will scale as $1/L^2$. One factor of L comes from the increased linear speed of the u - v tracks (per wavelength) due to rotation of the earth, and the other comes from the increased radial spacing between adjacent tracks. If the sampling density in the u - v plane is held constant between two configurations A and B , the ratio of the observation times should be $(L_A/L_B)^2$, with the larger array observing longer. A similar criterion arises from the specification that the signal-to-noise ratio should be the same for each array imaged separately, when viewing

perfectly smooth emission. The area¹⁸ of the synthesized beam scales as $1/L^2$ and so also does the flux density per synthesized beam. The nominal thermal noise does not change with scaling, and again we recover a ratio of $(L_A/L_B)^2$. Both of these criteria are approximations, of course. The array configurations do overlap quite significantly, and the actual response to extended structure is very much a function of that structure. For nonlinear issues such as deconvolution, one might wish to concentrate measured visibilities in regions where the source visibility function is changing rapidly, or one might wish to use less time in regions that will later be downweighted by the restoring beam. But in rough terms one would desire the square of the scale size ratio. For adjacent configurations of the VLA, the squared scale size ratio is 3.2^2 or 10.2 . This rule would lead to a ratio of more than 1000 in observation time between the D & A configurations! In practice, one never has anything like this ratio to work with. Typically, one might have one or several hours in the smallest configuration, several times that in the next largest, and “as much as one can get” after that. Very often multiple configuration data sets are combinations of archival data taken for different projects. Multiple configuration observations are hence not easily typified, because they are dominated by pragmatic concerns. If there is a typical 4-configuration data set, however, this one might be characterized as having more high spatial frequency data than typical, but still less than would be desired from the sensitivity concerns.

The basic point to take from this simulation is that the same regimes as usual are present, and that the difference between uniform and natural weighting is even larger than for a single data set. If ad hoc reweighting constants are introduced between data sets, the thermal RMS will change and in most cases likely get even worse.

The uniformly weighted beam has an RMS/resolution of $3.11/.085''$. At a robustness of -0.5 , this becomes $2.25/.092''$, with almost no effect on the effect on the beam. (The $FOV_w = 1/4$ super-uniformly weighted beam is $4.71/.071''$, and a 2048^2 pixel uniformly weighted beam is $2.49/.094''$.) At a robustness of 0 , the RMS has dropped by a full factor of 2 to 1.56 , though the beamwidth is now $.126''$. The beam has grown a bit of an extended shelf, though at a level which would likely not impede the deconvolution. At a robustness of 0.5 , the parameters have become $1.13/.202$. The far-sidelobe level is fairly good, though the main lobe has a somewhat peculiar shape. The minimum sidelobe value of robustness is found somewhere between 0 & 0.5 , and beams in this range might be reasonable for projects that need high sensitivity. Certainly the 0.5 robustness beam is in every way superior to the naturally weighted beam, which is $1.00/.285$ and has a tremendous extended shelf.

¹⁸The area of a synthesized dirty beam is of course zero when integrated over the entire dirty beam. This use can be considered an approximation when looking at smooth emission with modest support. Alternatively, almost no matter what criterion one uses to determine the effective area of the dirty beam, the effective area will certainly scale as the inverse square of the array size.

Finally, it should also be mentioned that if one is willing to sacrifice significant resolution, multiple configuration observations are particularly amenable to beam shaping by tapering. This data set is used again in Figure 3.28 of Section 3.8.4, where a beautiful (low resolution) Gaussian beam is produced without use of robustness.

3.7.6 AT Full Tracks (SN1987A)

Observational data of SN1987A courtesy of Lister Staveley-Smith. 2 configurations of the Australia Telescope Compact Array, 3.7 hours of data spanning 5.7 hours in configuration 6C on 21 October 92, 7.7 hours of data spanning 8.8 hours in configuration 6A on 4/5 January 93, 8.64/8.90 GHz, -69.3° , 15 second integrations, very good u-v coverage (nearly full tracks) but with missing wedge and isolated points, system temperature on the furthest antenna out is half that of the others, very high oversampling

This is the data used in Chapter 8, and in the striping demonstration of Section 3.5. Exactly the same regimes are present as in the other good coverage cases. The uniform RMS/resolution goes from $1.73/.897''$ to $1.33/.912''$ at a robustness of 0, and to $1.10/.969''$ at a robustness of 0.5. The naturally weighted beamwidth is $1.132''$. Significant sensitivity can be gained by working with moderate robustness, but the low sidelobe compromise beam is essentially not present. A robustness of 0.5 is superior to the naturally weighted beam, and nearly as sensitive, but sidelobe structure leaves something to be desired. The missing wedge of data is characteristic of east-west array observations with less than 12 hours of coverage and has generated a fairly eccentric beam. No single value of the robustness can compensate for the near-in sidelobes at all position angles simultaneously. At the robustness where the near-in sidelobes along the major axis pass through zero, the minor axis still has a significant first negative. At the robustness where the minor axis goes through zero, the major axis has acquired a significant shelf. An experimental azimuthally varying version of the robustness is presented in Section 3.12, but this is only partially successful — certainly more work could be done with the concept. The periodic sidelobes are a consequence of the east-west geometry of the array and the regular spacing interval between the ATCA antenna stations. No weighting scheme is likely to significantly alleviate the presence of the grating rings.

It is difficult to control the first sidelobes of asymmetric beams with a single value of robustness. The RMS/resolution tradeoff is still useful.

3.7.7 VLBA Full Tracks

Synthetic data produced by SDE viissim, VLBA, 1.66 GHz, 50° declination, 30 second integrations, full tracks (horizon-to-horizon), excellent u-v coverage, moderate oversampling.

The VLBA antenna locations were chosen to optimize imaging performance of the array, subject to a number of logistical constraints. This is reflected in the excellent

properties of the uniform synthesized beam. In terms of sidelobes, it is difficult to imagine an array of this nature doing substantially better, and the low RMS degradation of 1.65 for uniform weighting reflects the care taken in array design. Possibly partially due to this optimization, the coverage responds particularly well to robustness near the uniform end of the curve. The change in thermal RMS is not dramatic, but the beam itself is almost unchanged. A robustness of zero produces a beam of fitted beamwidth close to uniform (4.96 mas vs. 4.74 mas), but with somewhat improved thermal RMS, (1.35 vs. 1.65). The already fairly soft negative sidelobes are further softened slightly. As might be expected from the comparative sparsity of the u - v coverage and analogy to the VLA snapshot simulations, this case also responds well to gridless weighting. The gridless uniform RMS/resolution is 1.31/4.80 mas, which becomes 1.26/4.96 mas at a robustness of zero. This beam is not shown, but it is visually nearly identical to the robust=0 panel in Figure 3.20. This is a good choice for medium to high resolution mapping. As the robustness is increased further, the central plateau in the beam lifts off before the noise drops substantially. By a robustness of 0.5, the plateau is already too wide to recommend, though certainly much improved over natural weighting. This is a direct consequence of the modest negative sidelobes to start with. As an alternative to the gridless weighting, it is useful here to super-uniformly weight. The first sidelobes go considerably more negative and are then brought back to near zero with robustness. With $FOV_w = 1/4$, and a robustness of 0.4, we obtain a similarly nice beam with an RMS of 1.20, a mean width of 5.16 mas, and improved sidelobes. The beam is again not shown, but is visually very similar to same panel as before. For all but the highest resolution projects, either of these two choices are good general purpose beams.

Mild robustness or gridless weighting works well for the VLBA case. Improvements are modest, but at almost no cost in beam shape. There is no particularly attractive medium resolution beam.

3.7.8 VLBA/VLA/GBT Full Tracks

Synthetic data produced by Caltech package `fake`, VLBA + phased VLA + Green Bank Telescope, 1.66 GHz, 50° declination, 60 second integrations, full tracks (horizon-to-horizon), excellent u - v coverage, moderate oversampling.

This simulation demonstrates the behavior of a badly unbalanced VLBI array, where the sensitivity of individual array elements varies greatly. The 10 VLBA stations have 25m diameter antennas, the Green Bank Telescope was taken to have a diameter of 100m, and the phased VLA was taken to have an effective diameter of 115m. As mentioned before, the snr weight scales as $D_1^2 D_2^2$, if all other considerations are held constant. These numbers are approximations and ignore considerations of receiver sensitivities and the like, but illustrate typical behavior. (Unfortunately, arrays such as this are all too common in practical VLBI.) The ratio of snr weights between the most sensitive and least sensitive baselines is 339! Consequently, the Fourier sum for the dirty map is completely dominated by the baseline between the two large elements. This baseline alone accounts for 45% of

the total snr weights. Baselines including at least one of the two large antennas makes up 96% of the total weight. Clearly then, the VLBA is providing the u - v coverage, while the two anchor antennas are providing the bulk of the signal.

Since robust weighting is based on the gridded snr weights, it works just as well when the nonuniformity in the gridded weights comes from differing snr weights instead of the gridding irregularities. For typical dense samplings, the probability of a significant gridding irregularity is reasonably smoothly distributed in the u - v plane and rises towards the outside of the sampling. This means that the effect of robustness on the large scale is similar to a taper. With an array where the snr weights are constant, the sidelobes smooth out uniformly over the beam as one goes from uniform to natural weighting. In the case of an unbalanced array, the “irregularities” that the robustness is correcting are distributed quite regularly, namely along the most heavily weighted baselines. Particularly in a sparse array, these tracks will be upweighted or downweighted largely as a unit when the typical gridded weight along the baseline is comparable to the robustness threshold. As the robustness is varied, the beam sidelobes change in a more chaotic manner than those of a homogenous array. This is due to the transform of individual tracks being added into the beam with changing weights. The thermal noise as a function of robustness has essentially the same behavior as before. The thermal noise is only a function of the weight histogram, and it is not sensitive to the u - v distribution of the gridded weights.

The RMS/resolution for the uniform beam is 4.98/4.82 mas, compared to the natural beam’s 1.00/9.19 mas. Clearly there is room for a weighting scheme to improve things! The gridless uniformly weighted RMS/resolution is 4.36/4.80 mas. Clearly it has helped, but not by a great deal. The fundamental problem with this data set is the varying snr weights, not the gridding irregularities, and there is little gridless weighting can do to help there. A robustness of -0.5 pulls the RMS down to 2.85, with a resolution of 5.54 mas. The sidelobe structure is essentially equivalent to that of the uniform beam, and this is a good choice for general purpose high resolution mapping. At a robustness of 0, the RMS is down to 1.66, and the beamwidth up to 6.57 mas. This would be a good weighting for projects needing high sensitivity. Beyond this point, the enormous sidelobe response of the VLA-GBT baseline becomes unacceptably high, and the beam unsuitable for most projects. Note that the slices in Figure 3.21 do not happen to cross the worst sidelobes. Between a robustness of -0.5 and 0, the highest positive sidelobe goes from 11.2% to 16.5%. This may or may not be acceptable for a given project, but the factor of 3.0 gained in thermal RMS over uniform weighting is a strong incentive to live with fairly high sidelobes. (The naturally weighted beam has a sidelobe of 32.0%, which is ridiculous given the superb quality of the u - v coverage.)

Gridless weighting doesn’t help much for unbalanced arrays. Robustness can help a lot.

VLBI practitioners often combat the difficult properties of VLBI beams by arbitrarily taking the square root of the actual snr weights. There are still some situations

where this might be useful, but in most cases robustness is a better approach. This is discussed further in Section 3.11.1.

3.7.9 VLBA/VLA/GBT/Orbiter

Synthetic data produced by Caltech package `fake`, VLBA + phased VLA + Green Bank Telescope, 1.66 GHz, 50° declination, 60 second integrations, full tracks (horizon-to-horizon), excellent u-v coverage, heavy oversampling.

Finally, we examine an extreme case of an unbalanced array. To the previous unbalanced array, we add an element similar to the upcoming Japanese satellite VSOP. This is a 10m antenna in an orbit¹⁹ designed to give good imaging performance. The simulated coverage is rather too good for a real observation, but again this is sufficient for an illustrative example.

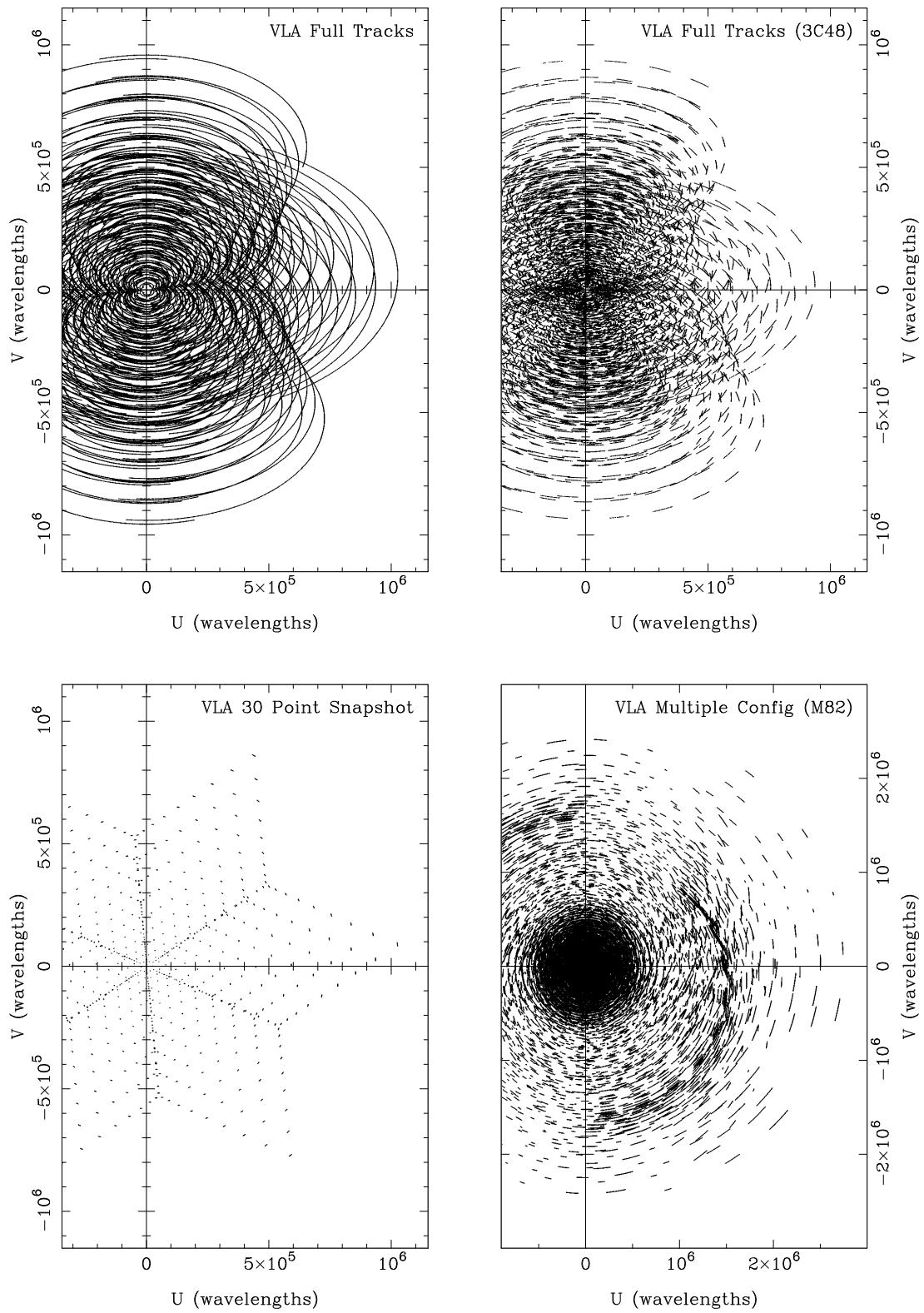
If the previous example allowed the weighting schemes some room to work, this is certainly more so! The uniformly weighted RMS/resolution is 18.63/1.21 mas, as compared to the naturally weighted 1.00/7.81 mas. The ratio of maximum to minimum snr weights is now 2116! The naturally weighted beam is essentially identical to that without the orbiter at all. Once again, gridless weighting is of little use — the gridless uniformly weighted RMS is 19.13/1.16 mas. This data set is unique in that gridless weighting produces a sharper beam than not. It is pathological in other ways as well. The high degree of oversampling was found necessary for the beam forcing algorithm of Section 3.11 to converge. (The algorithm was well behaved in all other cases studied.) Fortunately, robust weighting does work well with this data. A robustness of -1.0 yields a beam with RMS/resolution of 6.83/1.43 mas. Broadening the beam by only 17% has dropped the thermal RMS by a factor of 2.7, or the observing time by a factor of 7.4! The sidelobes of the beam have definitely become worse, both in terms of the near-in shelf and the far sidelobes. By a robustness of -0.5 , the beamwidth has broadened to 2.27 mas, and the RMS dropped further to 3.84. At a robustness of 0, the RMS is an appealing 1.87 but the beamwidth has more than tripled to 3.91 mas. By this point, equivalent data could be obtained with the earth bound array alone and there is little point in pursuing resolutions poorer than this.

Clearly one is paying a price both in resolution and in sidelobes for the considerable thermal noise benefit. It remains to be seen how much these sidelobes will affect the deconvolution. The sidelobes for robustness less than 0 are not that bad. Certainly for the compact sources likely to be imaged by orbital VLBI, it seems probable that the deconvolution will not be the limiting factor. This should be verified so far as possible in any given

¹⁹The actual orbital parameters used in the simulations were: semi-major axis = 15495140 m $\approx 2.43 R_{\oplus}$, eccentricity = 0.3864, inclination = 57.0°, ascending node = 5.0°, periapsis = 270°, mean anomaly = 0°, epoch = 0 UT on day 264 of 1986.

case, of course. If observations are proposed based on thermal noise calculations using robust weighting, more study and simulation would certainly be prudent. If one merely wishes to extract all the information possible from an existing data set of marginal quality, simply process the data with various beam shaping parameters, and use the resulting images as a guide in *post hoc* modelling.

Robust weighting is a big win for orbital VLBI — but be careful!

Figure 3.12a: u - v Coverages

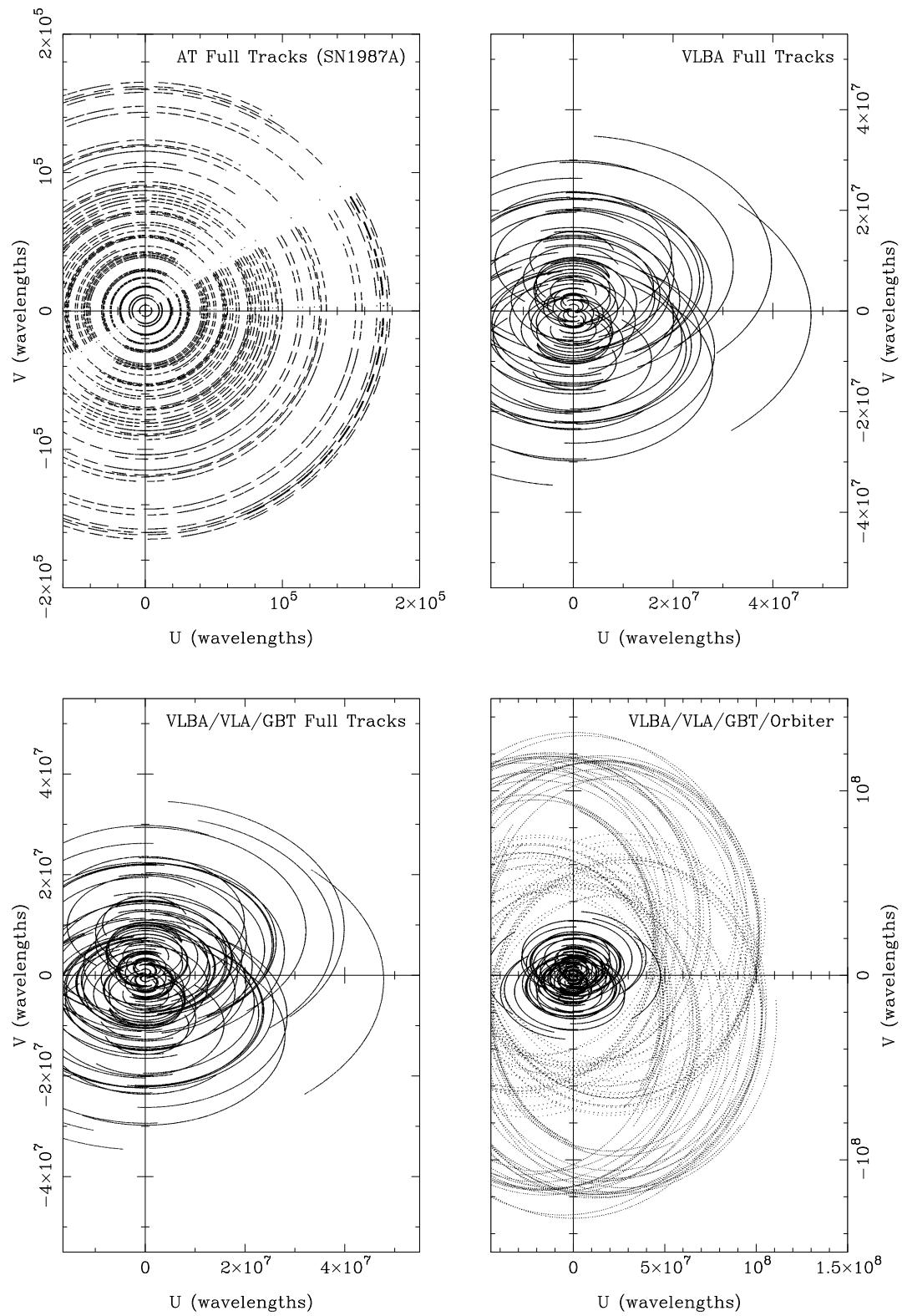


Figure 3.12b: cont.

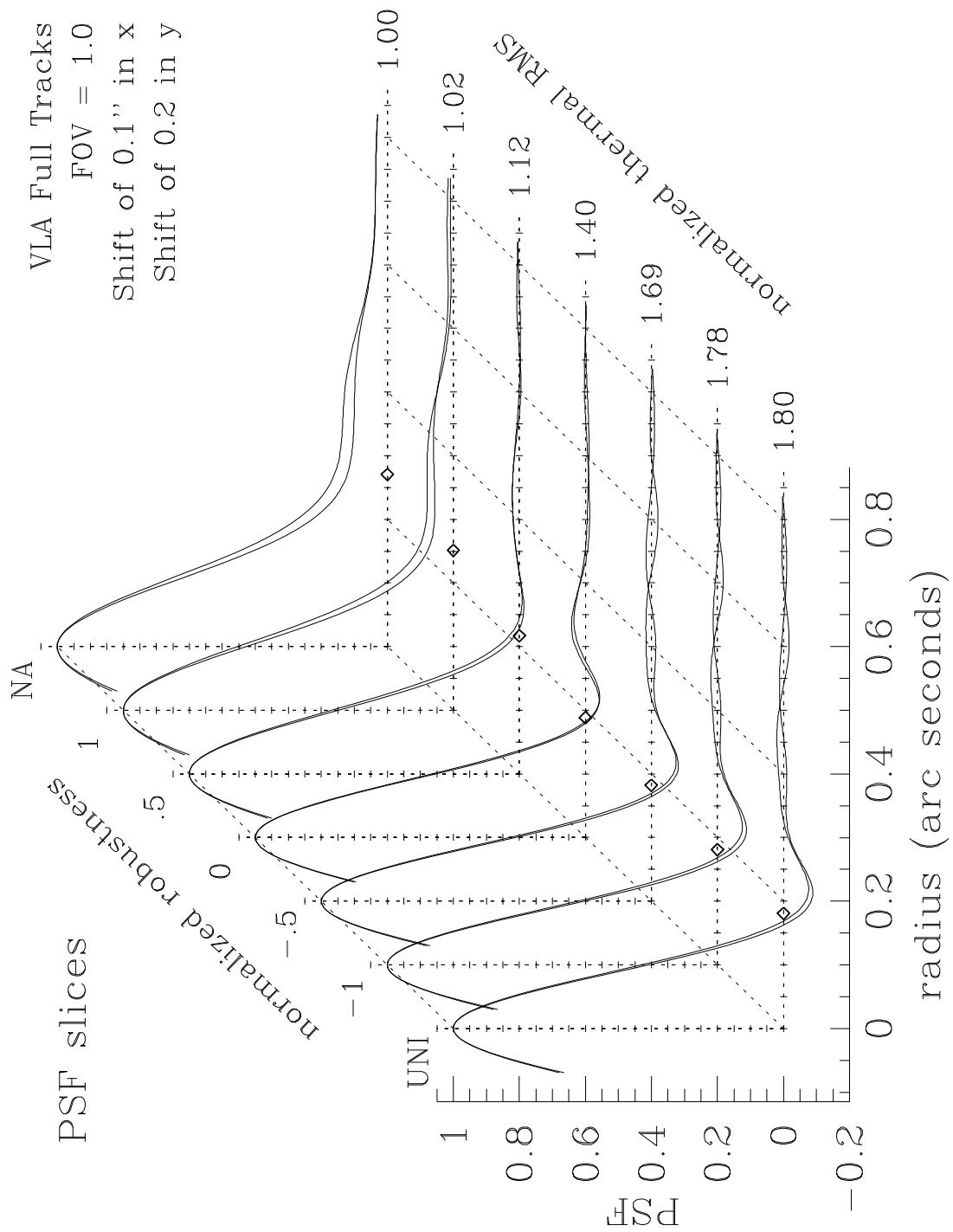


Figure 3.13a: PSF plots against robustness, VLA Full Tracks.

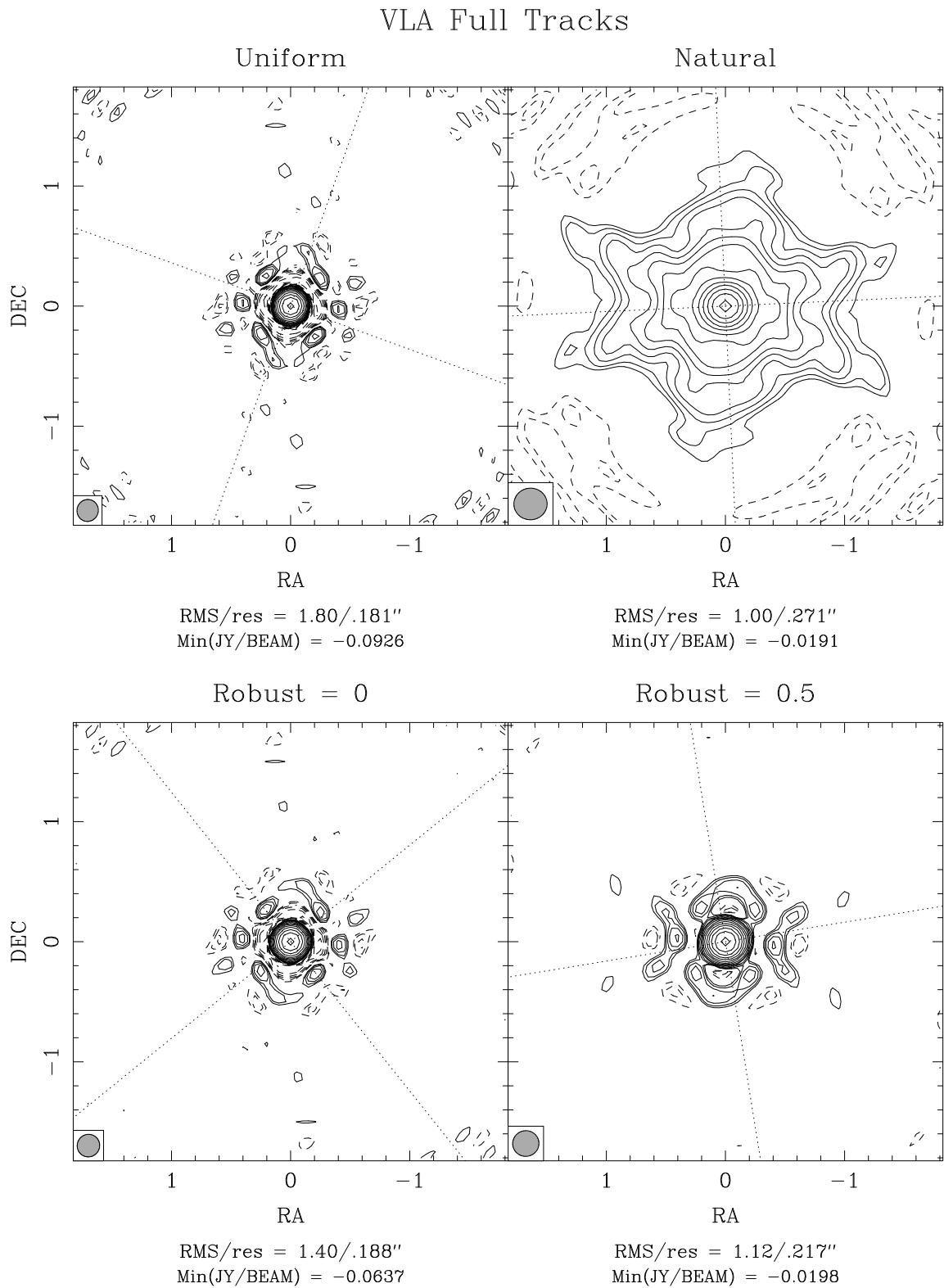


Figure 3.13b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

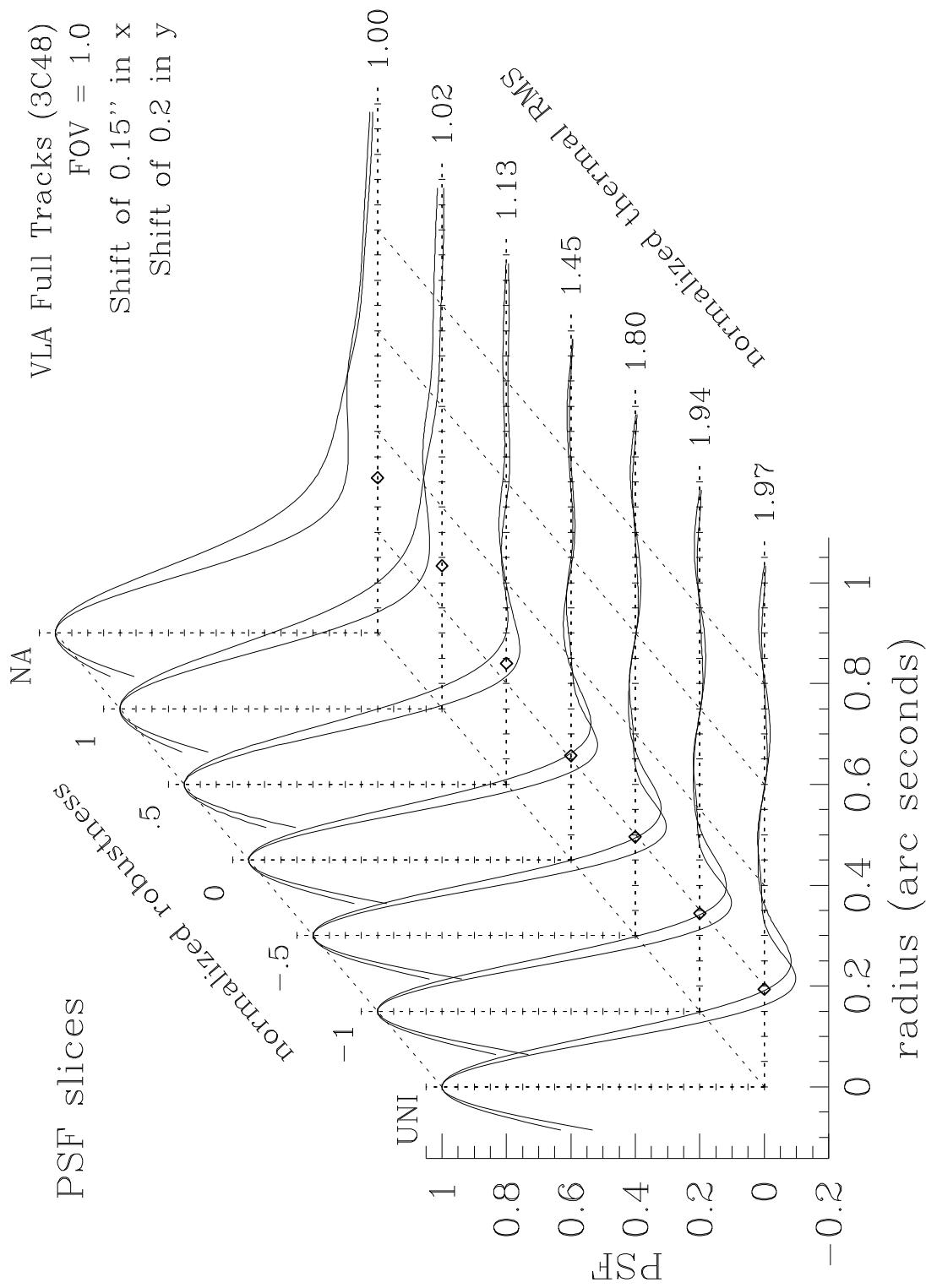


Figure 3.14a: PSF plots against robustness, VLA Full Tracks (3C48).

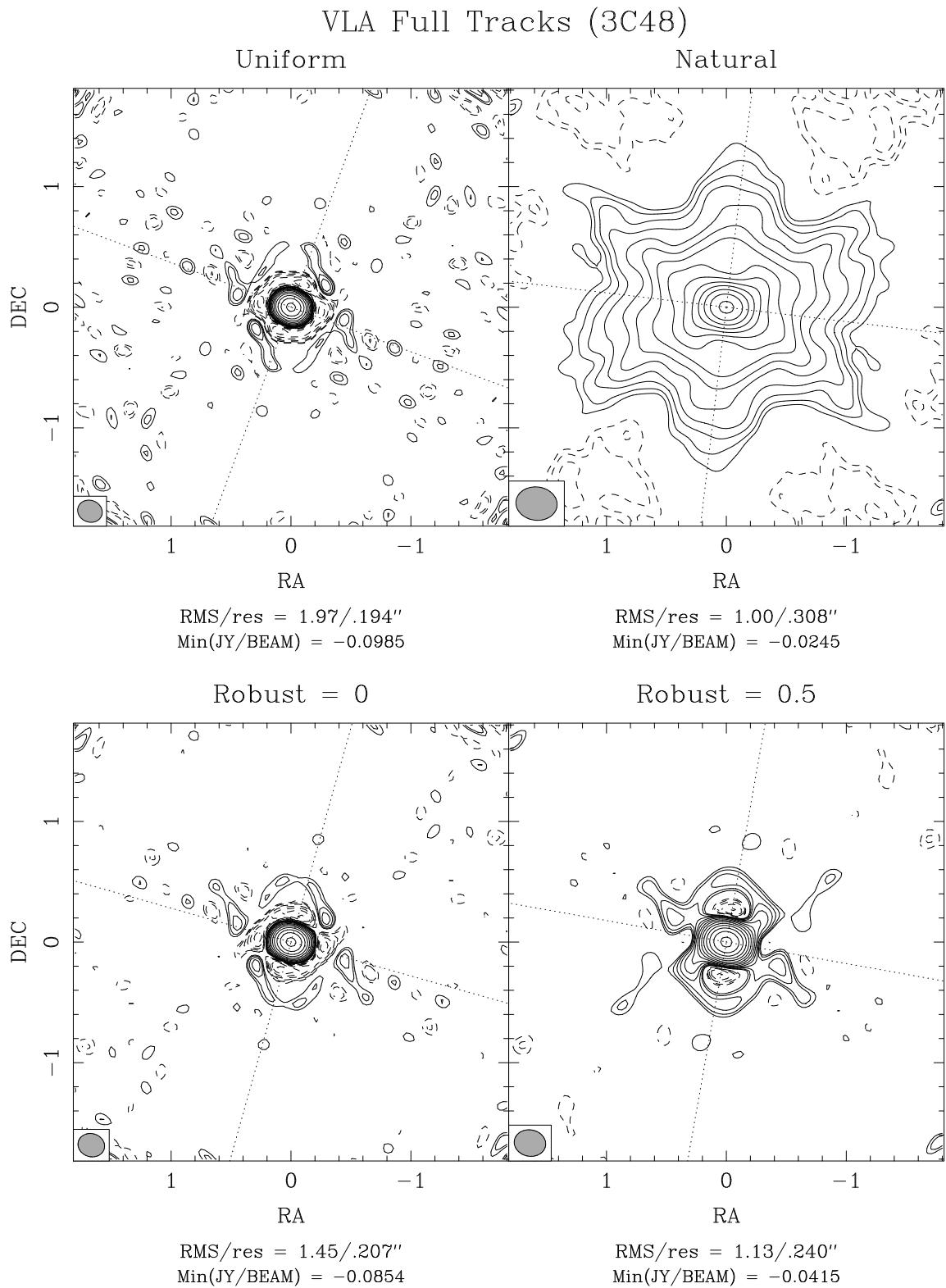


Figure 3.14b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

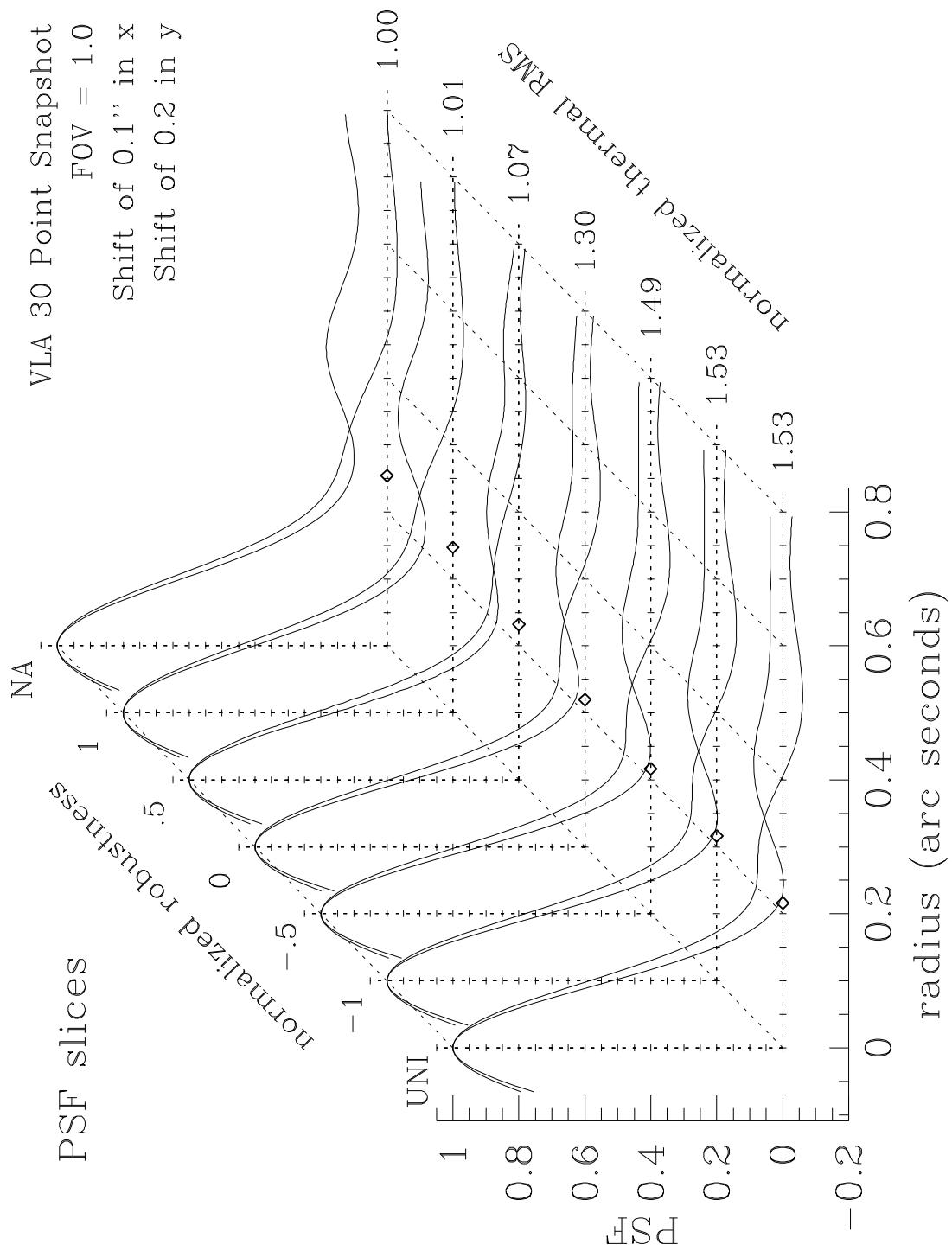


Figure 3.15a: PSF plots against robustness, VLA 30 Point Snapshot.

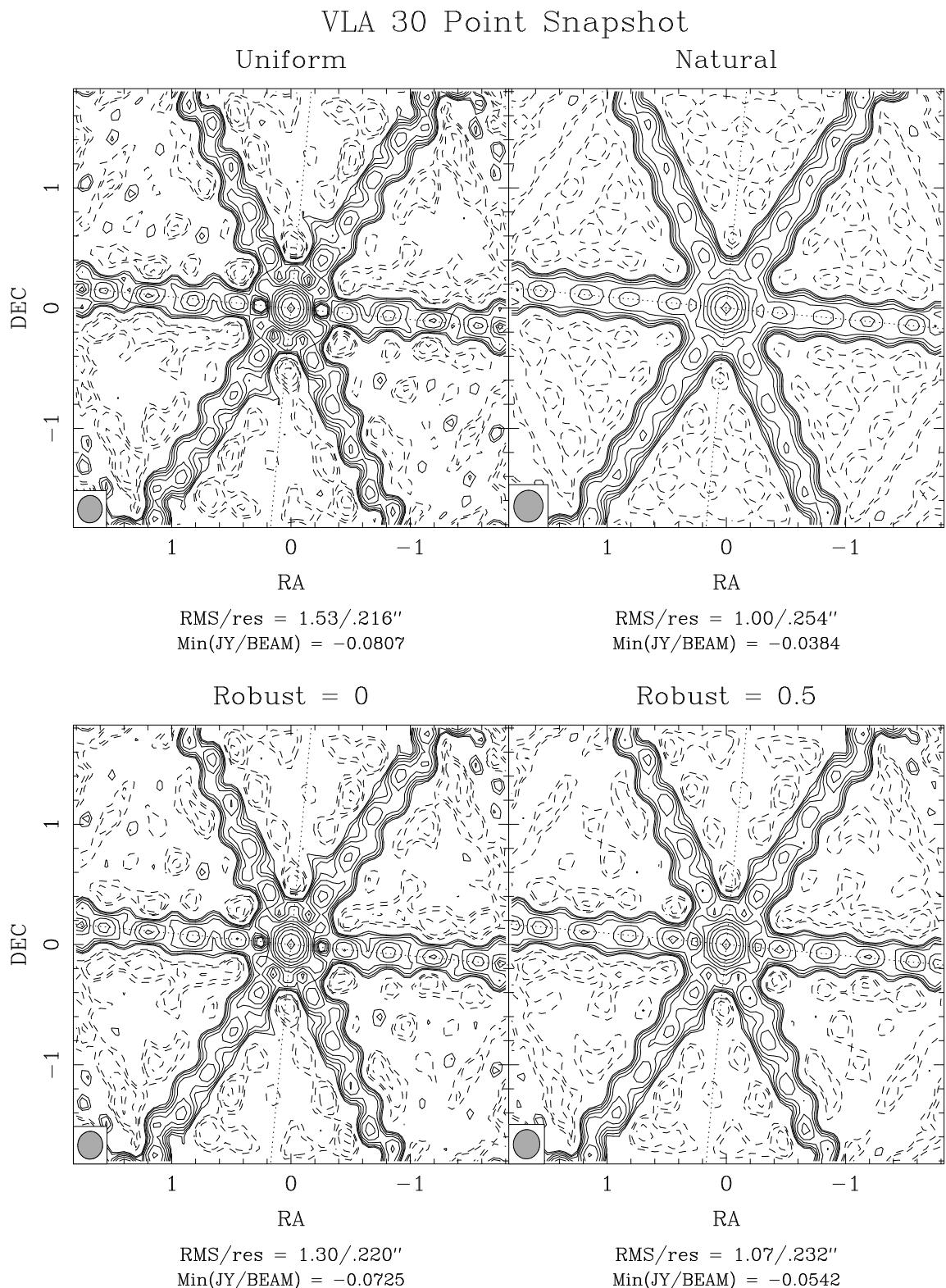


Figure 3.15b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

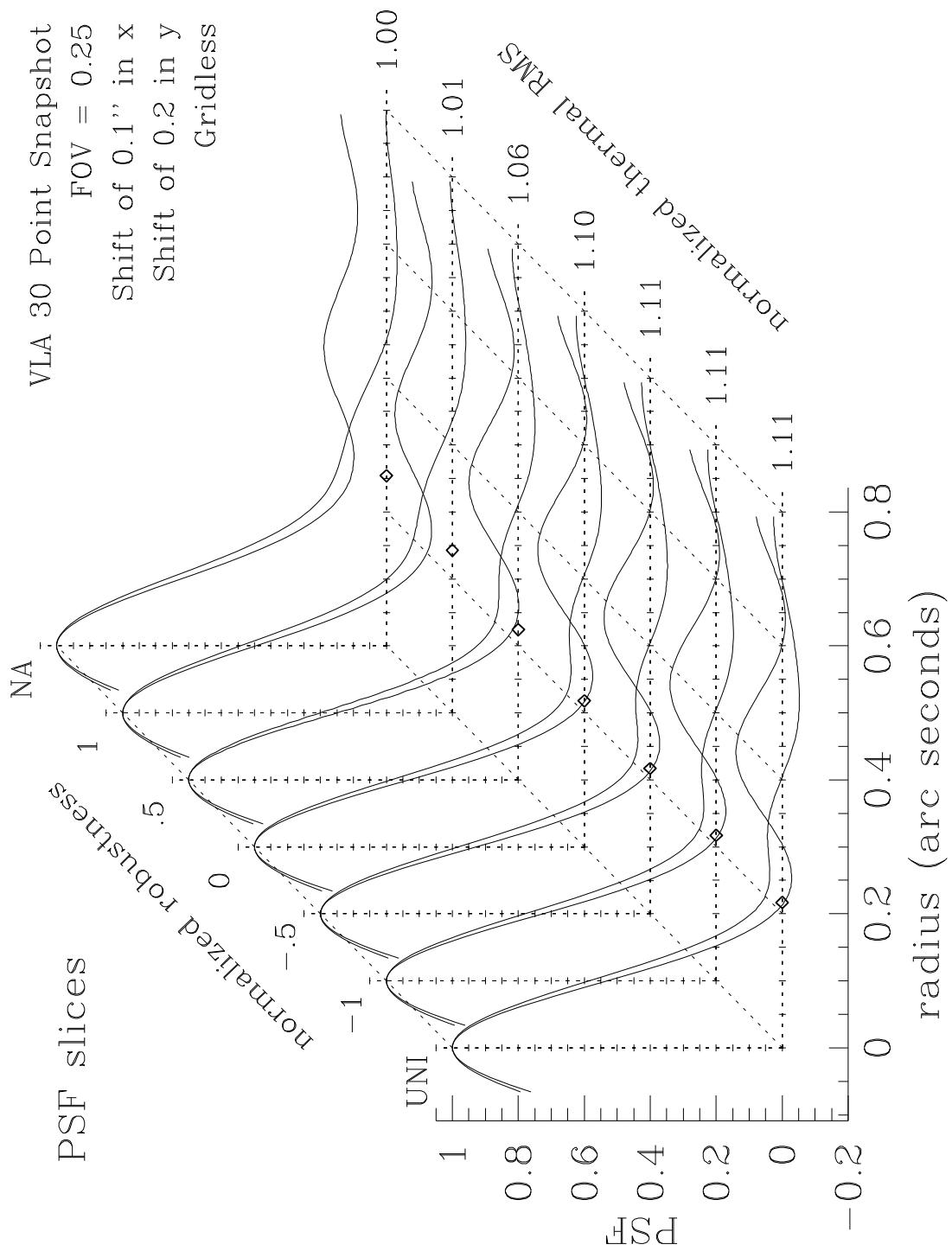


Figure 3.16a: PSF plots against robustness, VLA 30 Point Snapshot (Gridless, $FOV_w = 1/4$).

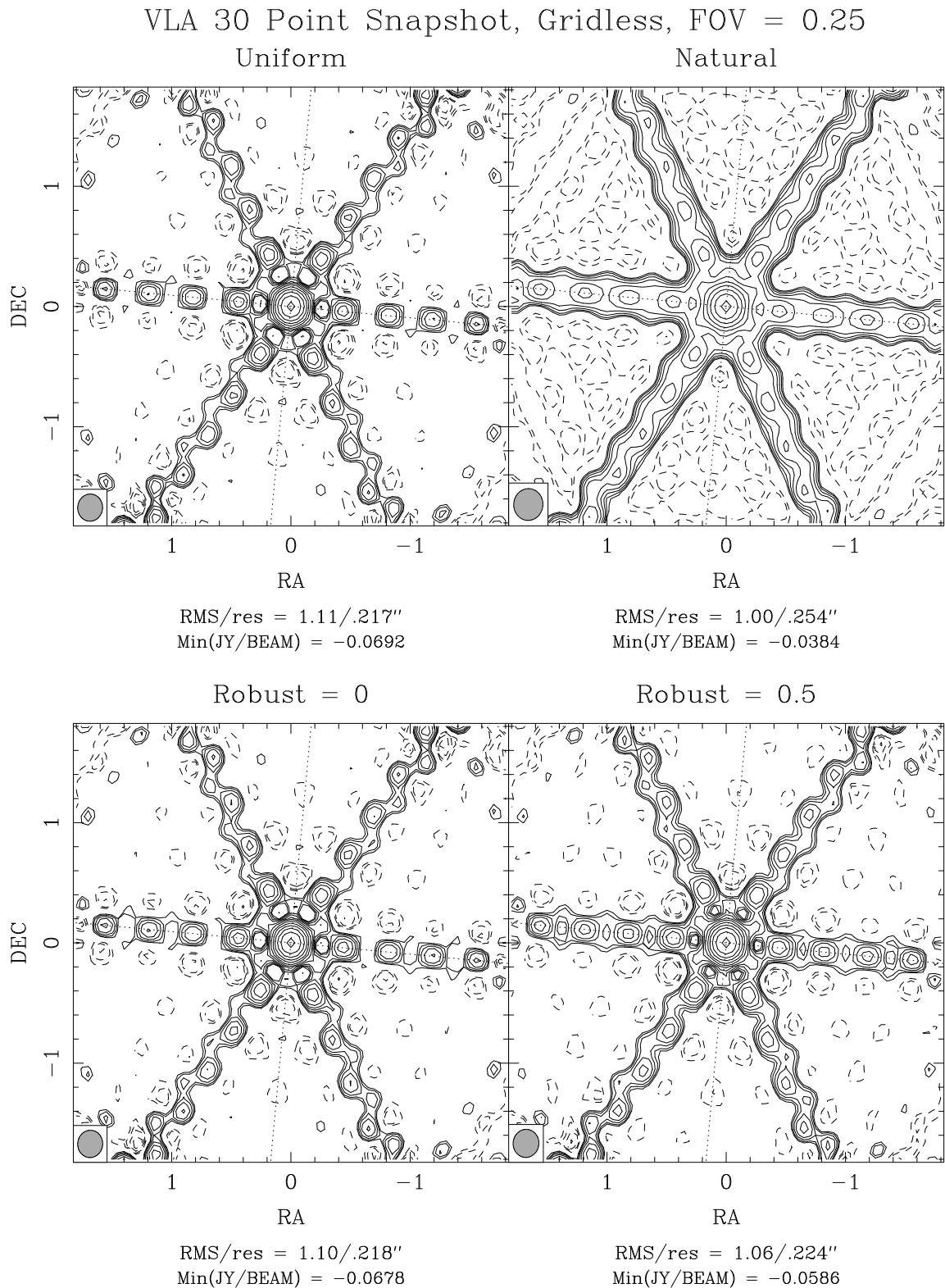


Figure 3.16b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

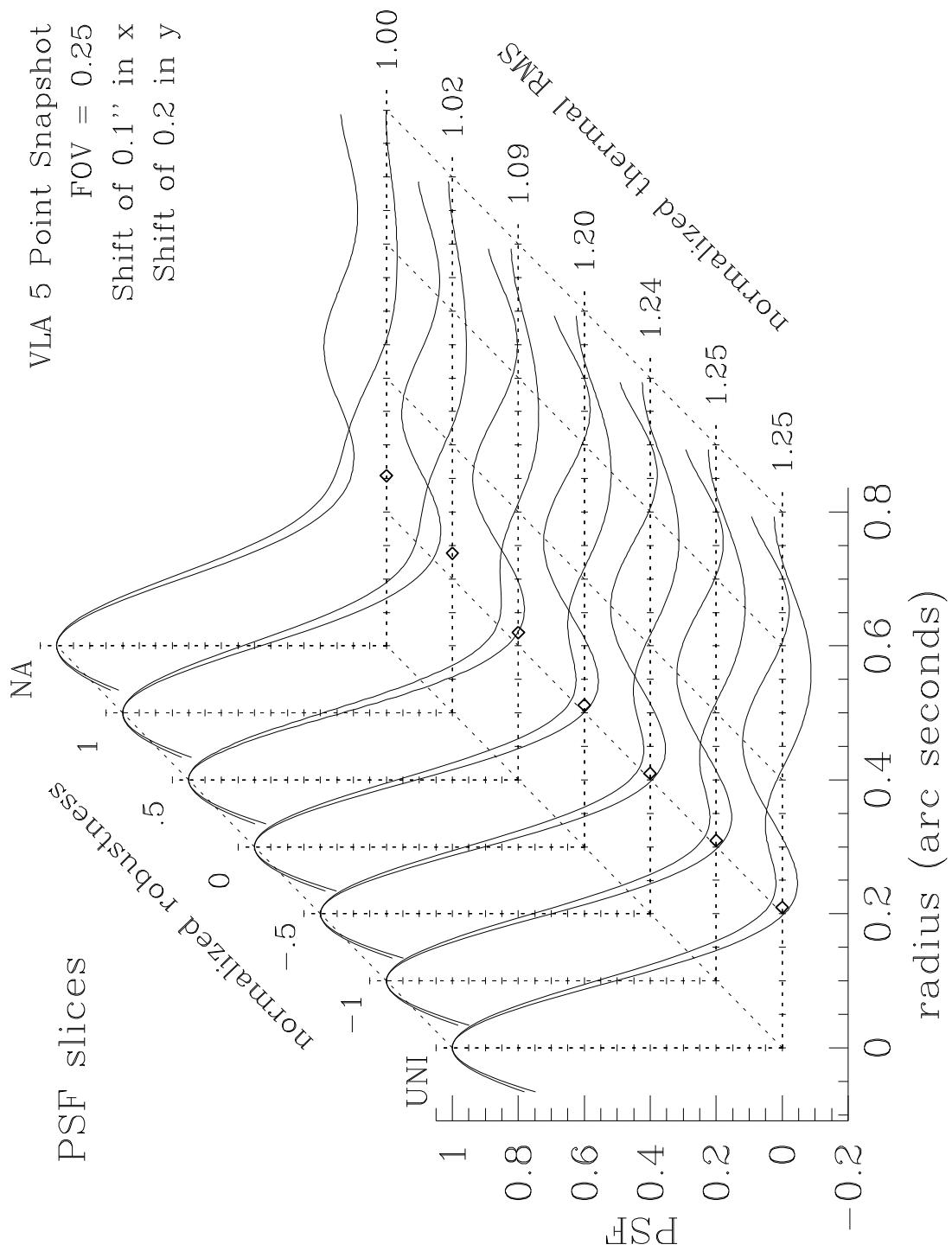


Figure 3.17a: PSF plots against robustness, VLA 5 Point Snapshot ($FOV_w = 1/4$).

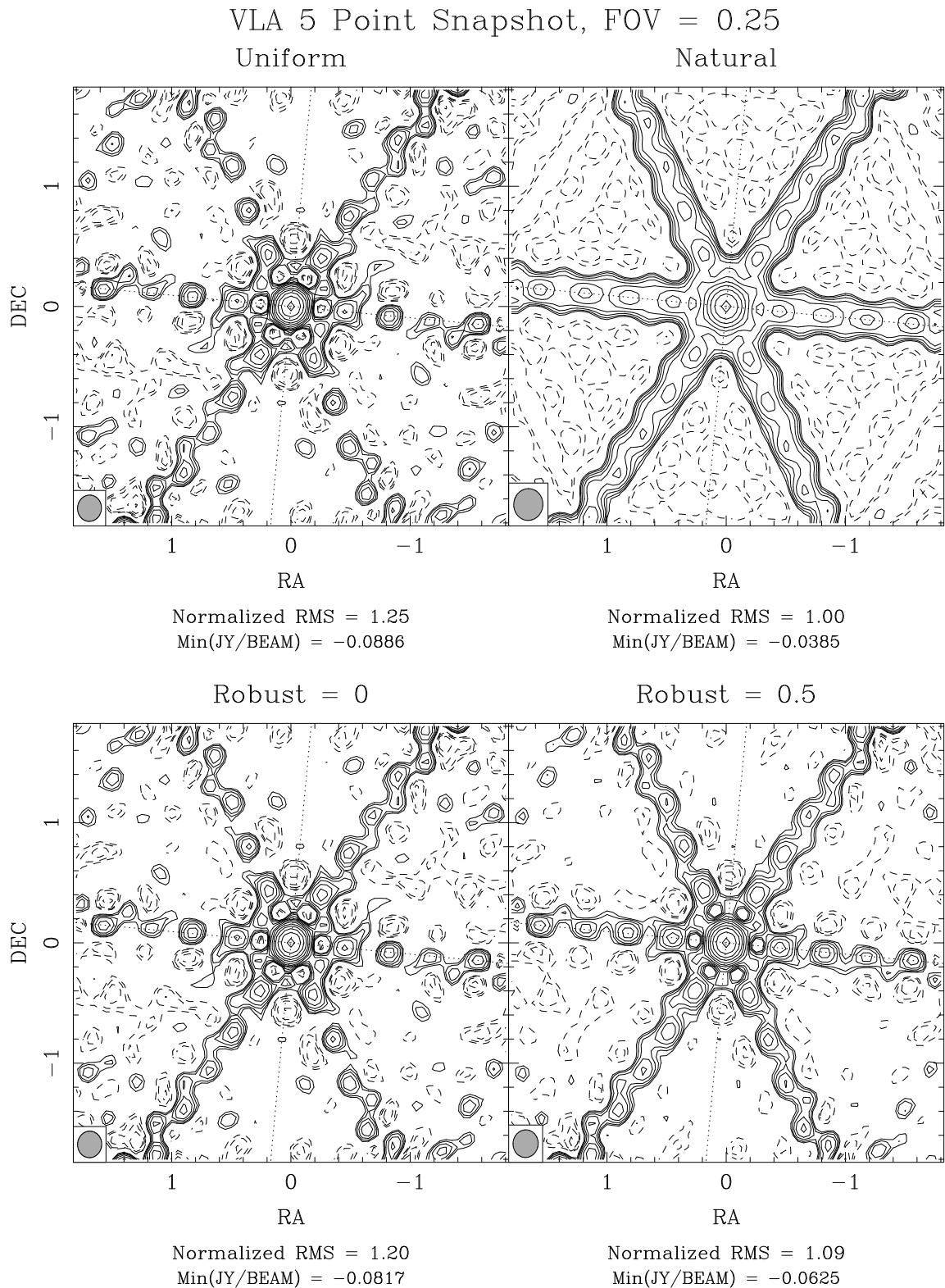


Figure 3.17b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

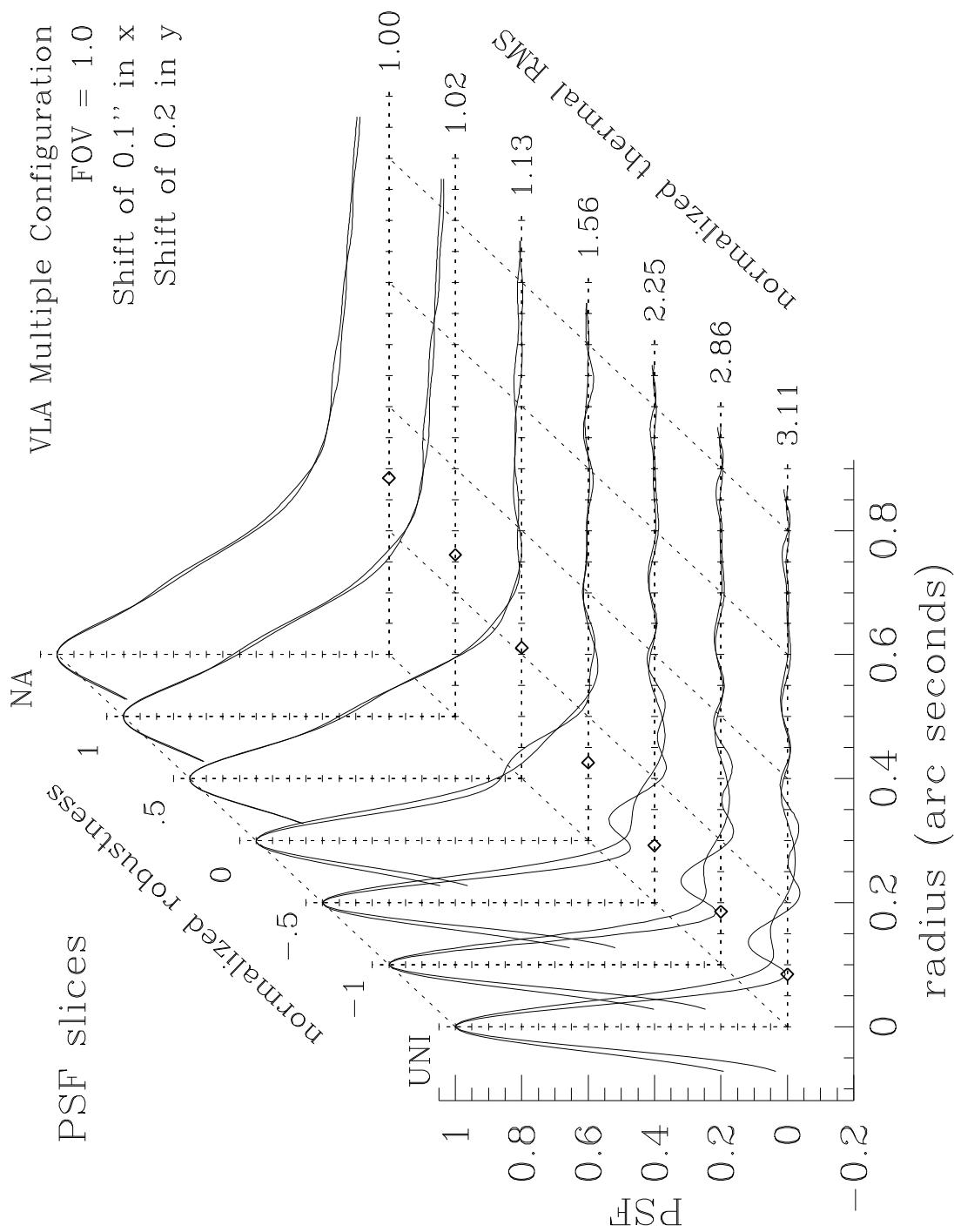


Figure 3.18a: PSF plots against robustness, VLA Multiple Config (M82).

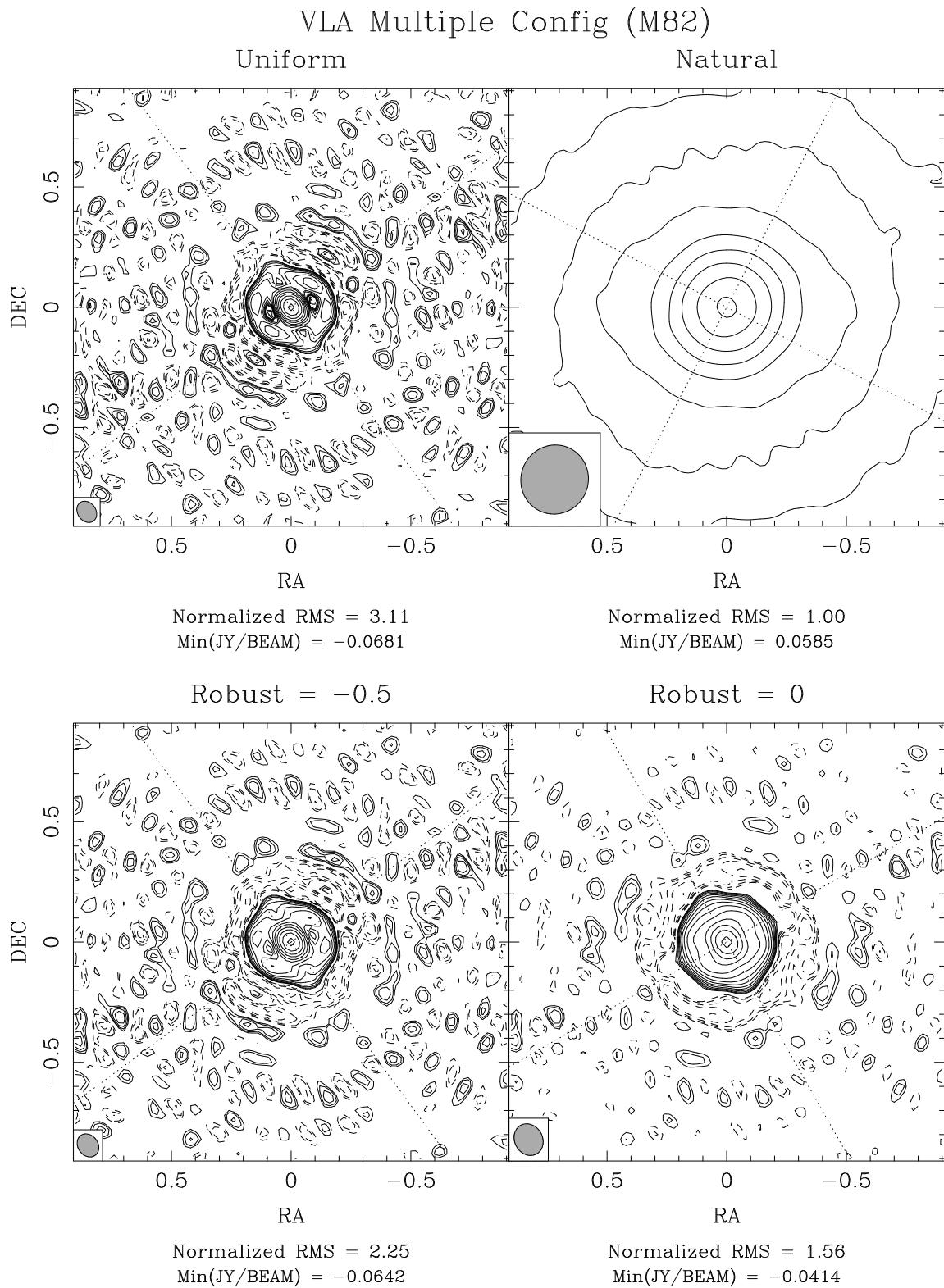


Figure 3.18b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

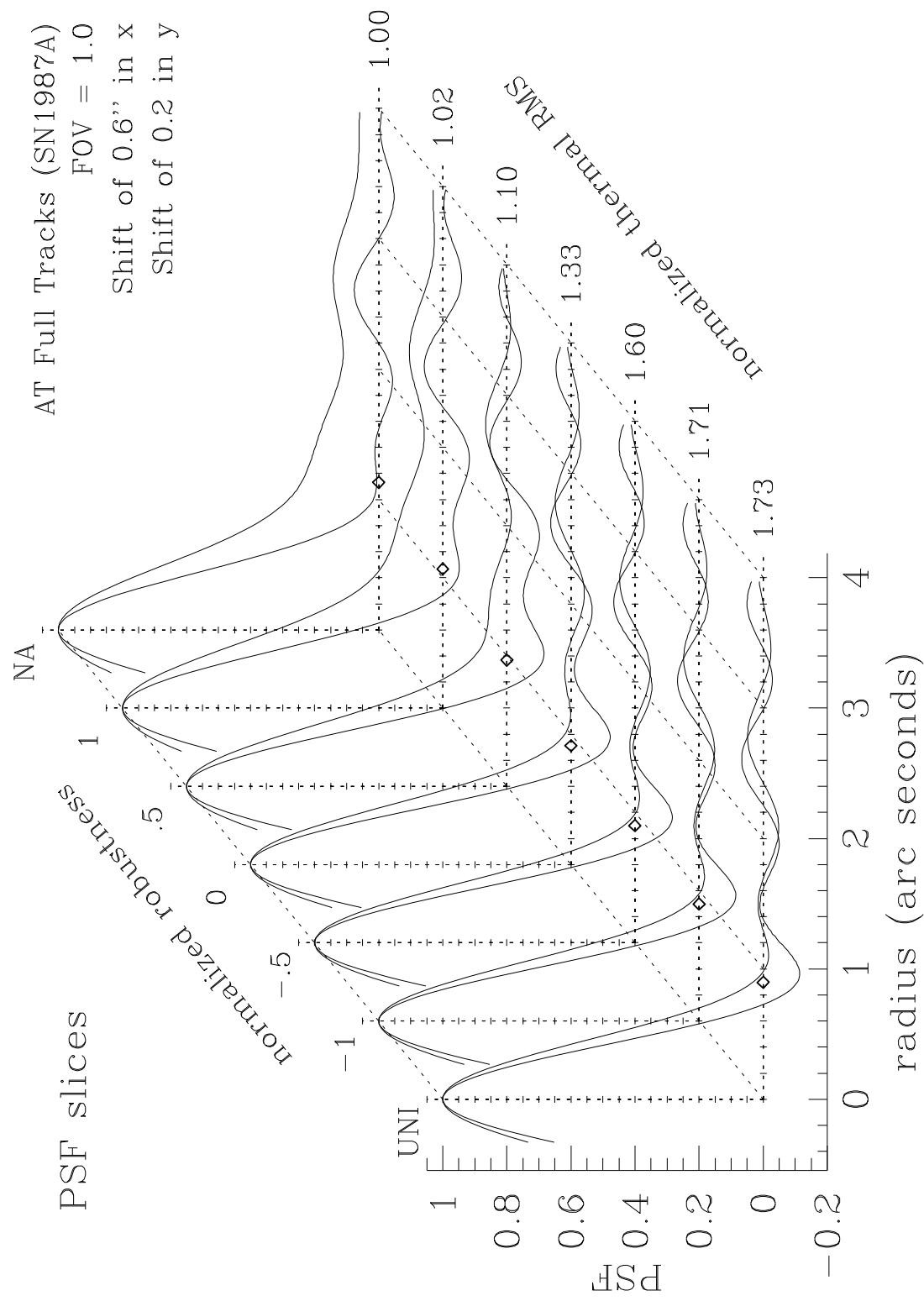


Figure 3.19a: PSF plots against robustness, AT Full Tracks (SN1987A).

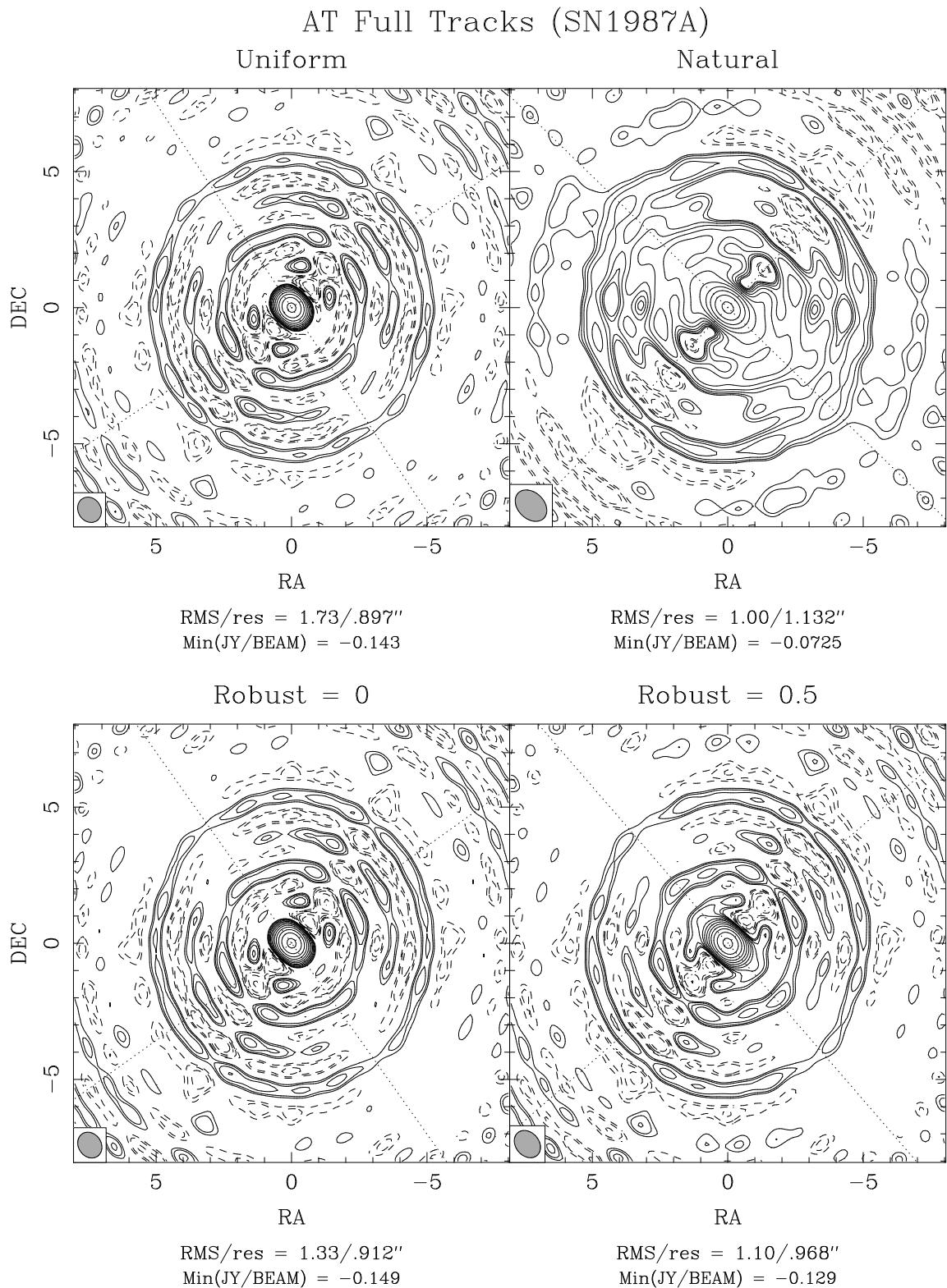


Figure 3.19b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

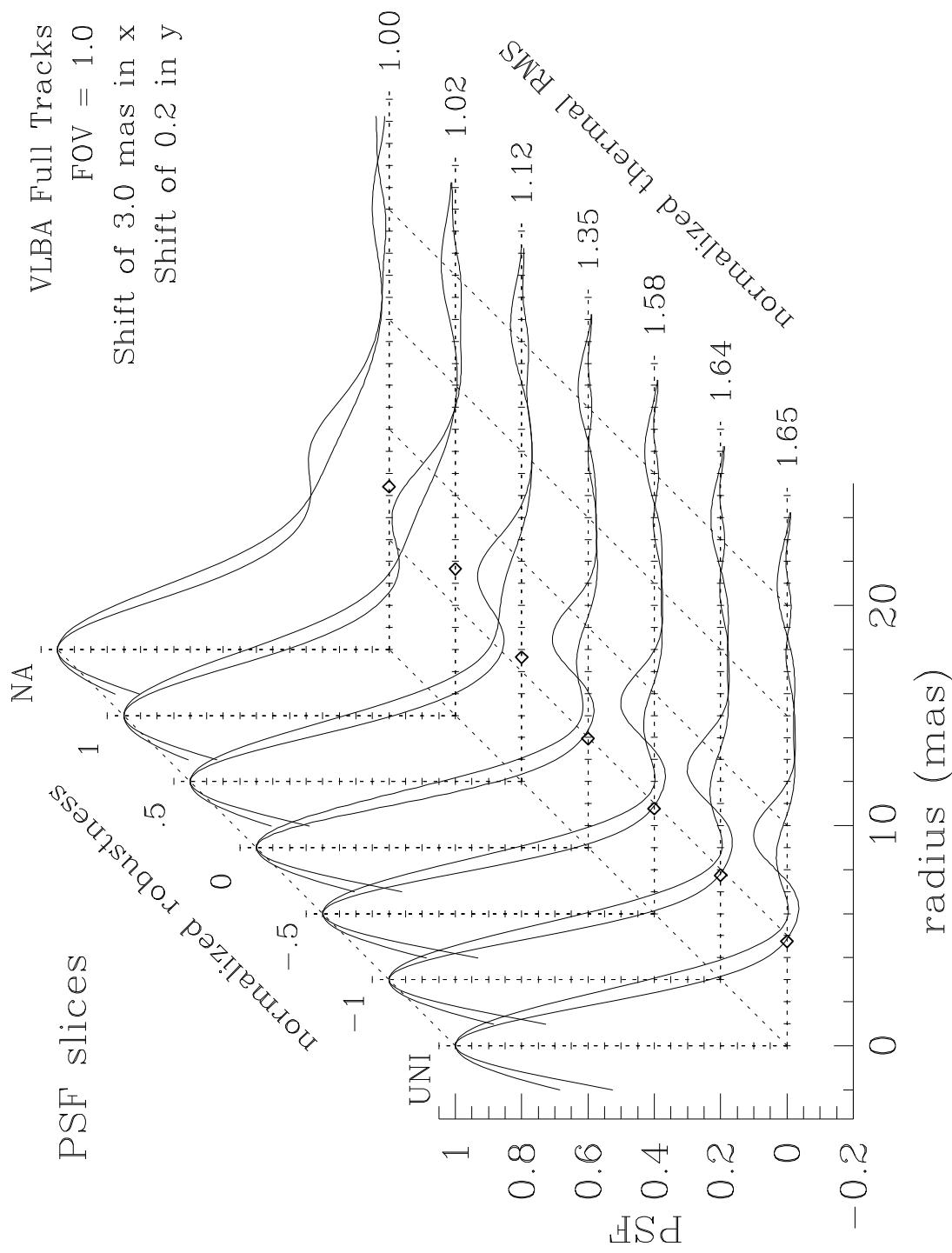


Figure 3.20a: PSF plots against robustness, VLBA Full Tracks.

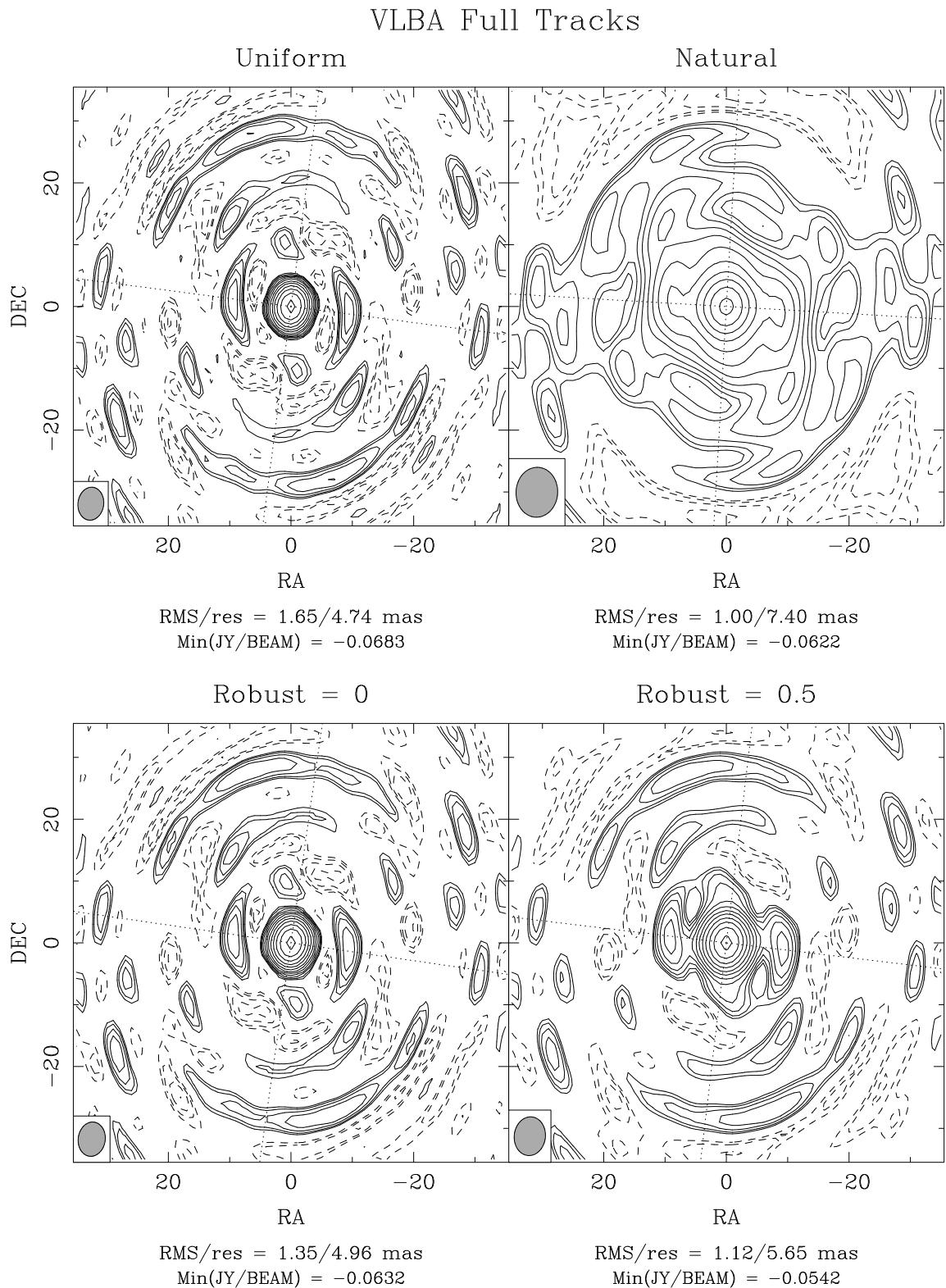


Figure 3.20b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

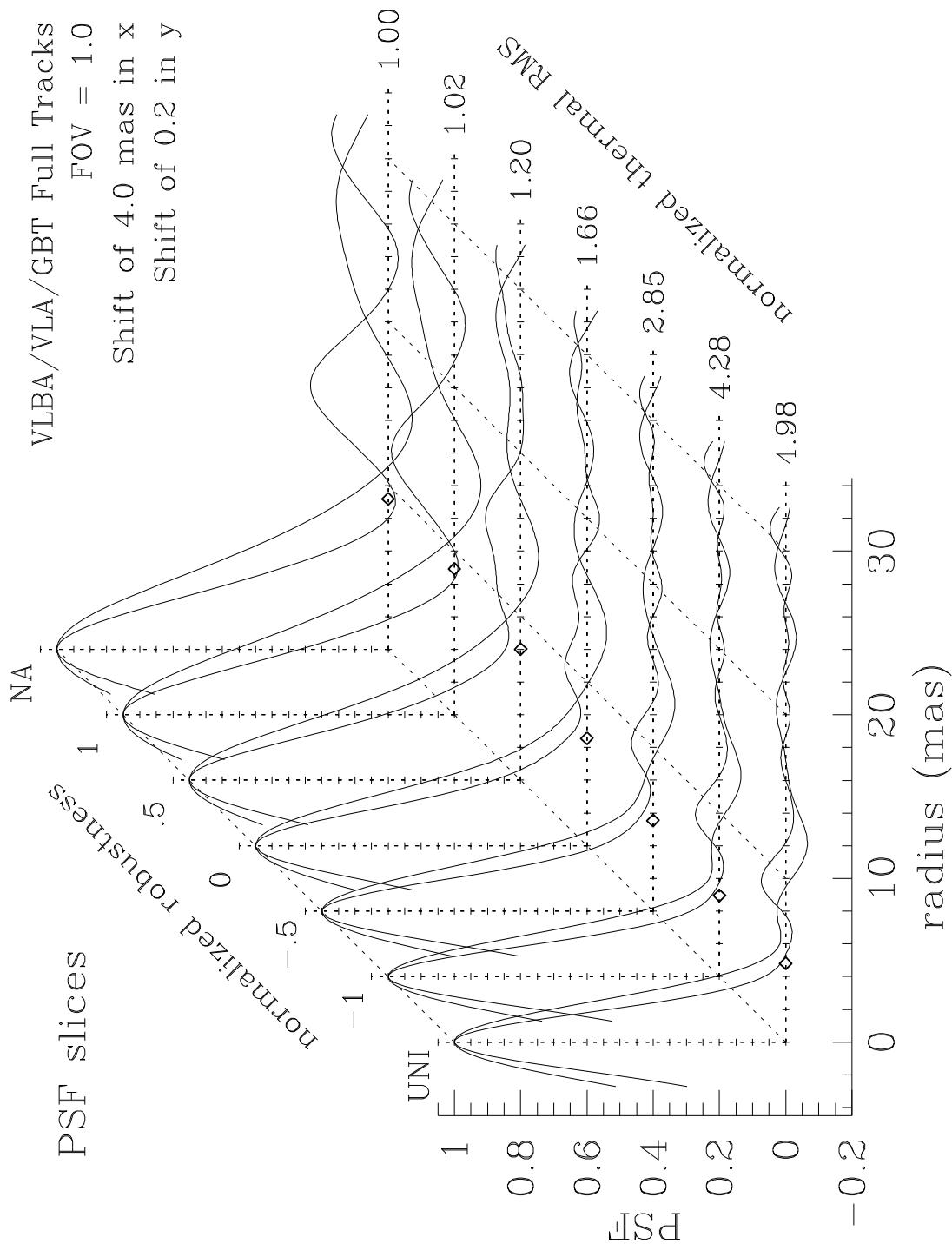


Figure 3.21a: PSF plots against robustness, VLBA/VLA/GBT Full Tracks.

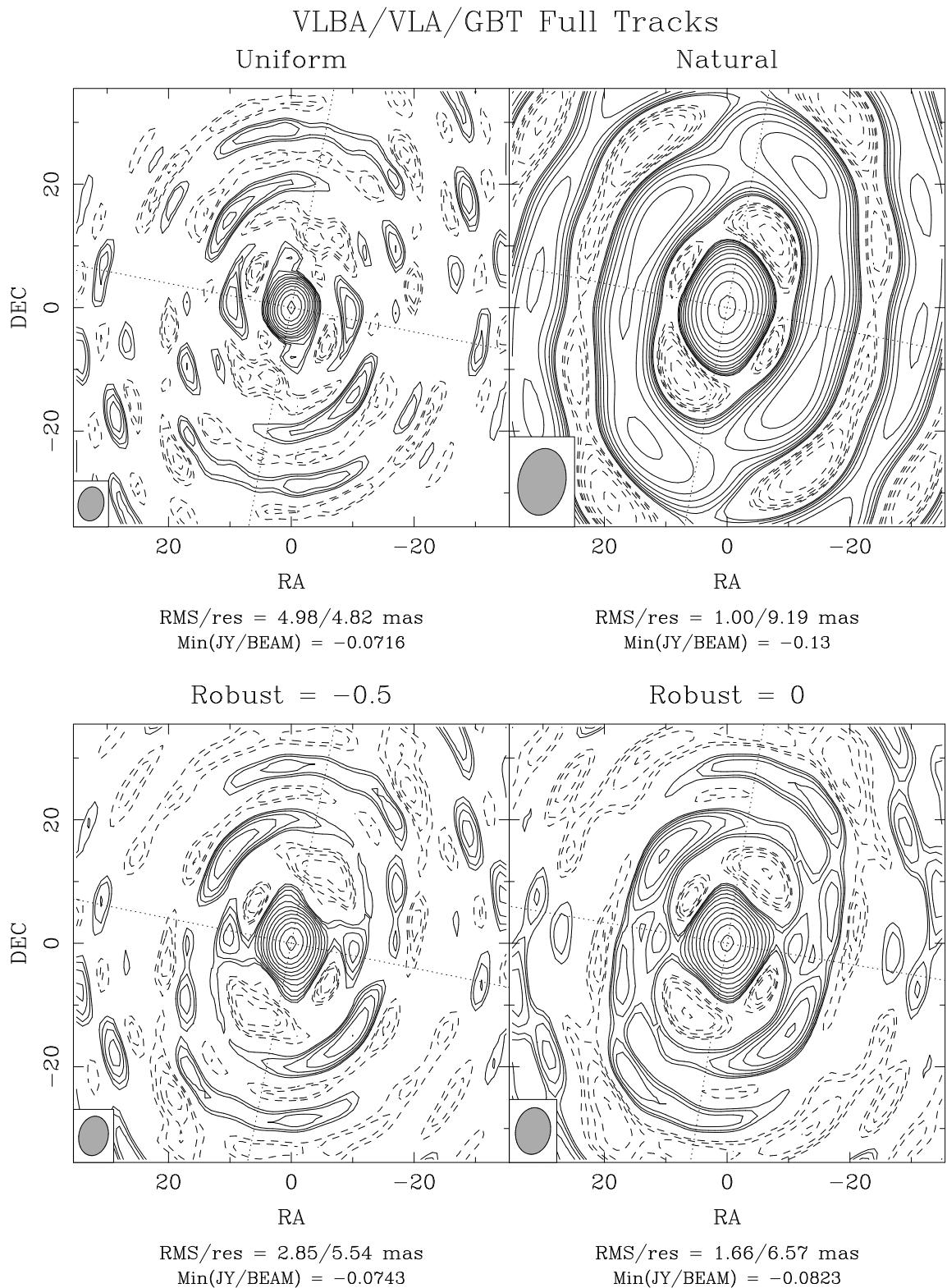


Figure 3.21b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

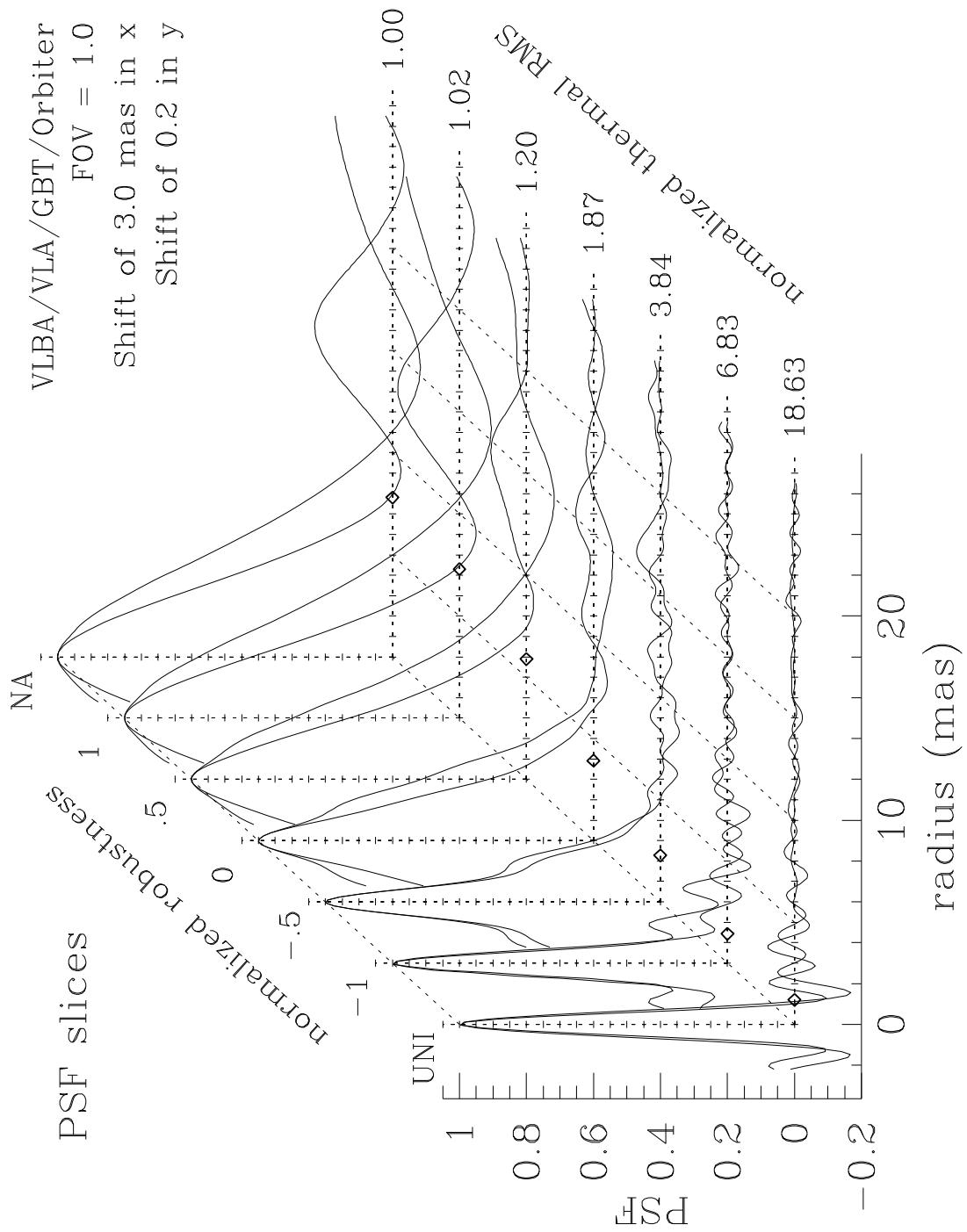


Figure 3.22a: PSF plots against robustness, VLBA/VLA/GBT/Orbiter.

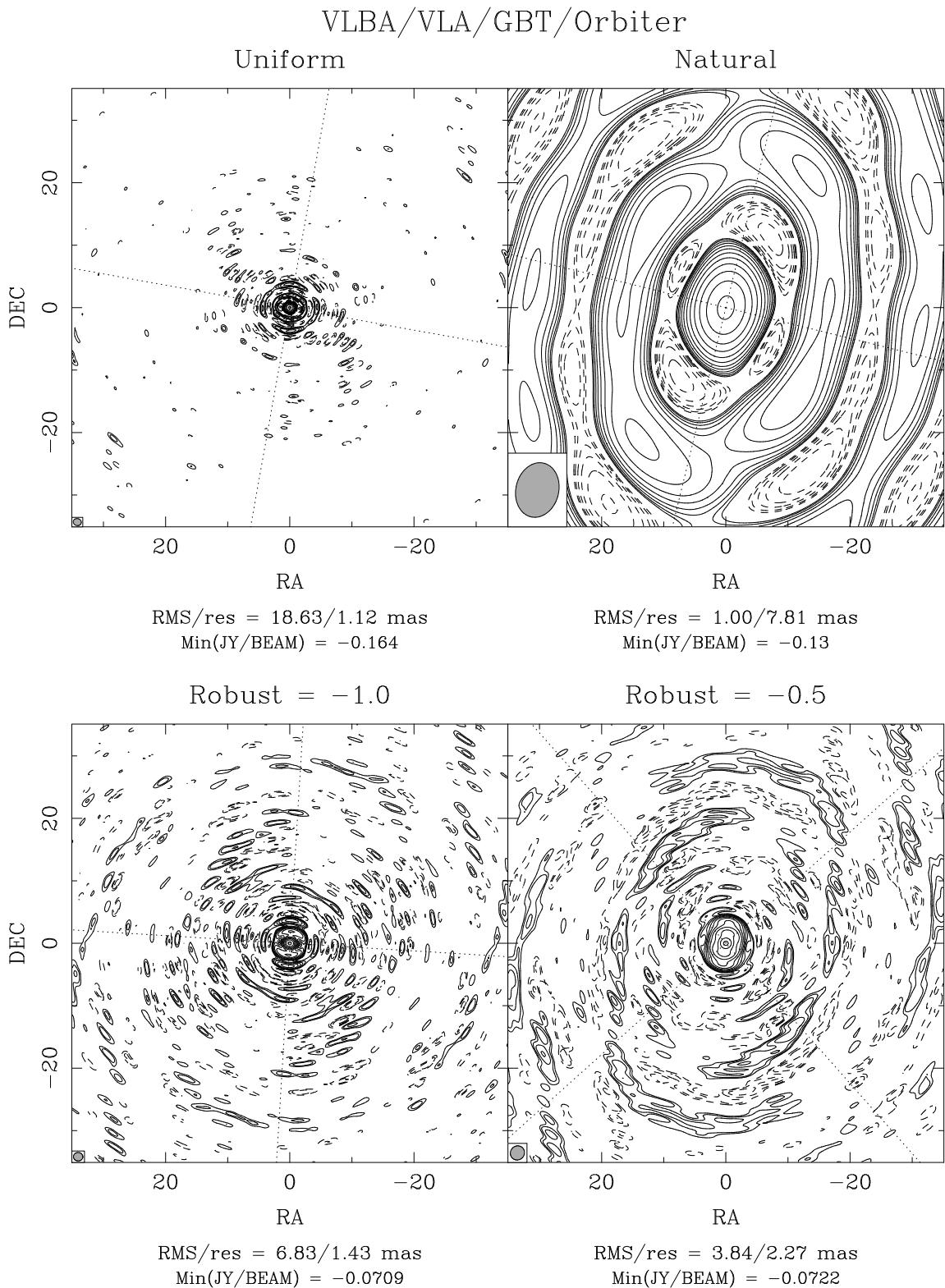


Figure 3.22b: Factor of $\sqrt{2}$ between contours, lowest is 2%.

3.8 RMS/resolution Tradeoff Curves

We now take a closer look at some RMS/resolution tradeoff curves as a function of different quantities. There are a lot of ways to get to the same place — the curves often look qualitatively the same when turning quite different knobs. But the whole point of this chapter is to produce the *quantitatively* best beam, and that is somewhat more difficult.

3.8.1 Tradeoff Parameterized by Robustness

The shape of the basic tradeoff curve parameterized by robustness is remarkably constant. Any given tradeoff curve, when plotted such that the uniform and naturally weighted limits are held at fixed points, looks quite a bit like any other. The interesting thing about these plots comes primarily from where the endpoints of each curve is relative to each other. Four curves are shown on each of the next few plots. There are two curves for two values of the weighting field of view, one at the normal uniform weighting and one super-uniformly weighted. The natural and (super) uniformly weighted limit of each curve is given as a diamond. Representative points along the curves are circled and the value of robustness shown next to it. Points along the curve are spaced at intervals of 0.1 in robustness. In addition, the same curves are presented using gridless weighting.

Figure 3.23 is quite typical for dense samplings. As with all of these plots, the naturally weighted limit is the same for all four traces. Super-uniform weighting on dense samplings produces a beam that has worse thermal noise and a tighter main lobe, moving the endpoint of the trace down and to the right on these plots. Since the curvature of the traces for different FOV_w is roughly the same, the super-uniform trace is below and behind the uniform trace. At a given value of thermal RMS, the super-uniform beam has a tighter main lobe. At a given value of beamwidth, the super-uniform beam has a lower RMS. The price one pays for this superior performance in the RMS/resolution tradeoff is sidelobe height. Typically the super-uniform beam will have somewhat higher sidelobes than the uniform beam at the same value of beamwidth. Still, this is usually a minor effect. For dense samplings, it often is a good idea to super-uniformly weight higher than normal, and then bring the RMS back to a reasonable value with robustness. Notice also that the gridless weighting curves are also somewhat “better” than their gridded counterparts, though the difference is slight. Gridless weighting makes a considerable difference when compared to the gridded uniform limit, but the gridded, $FOV_w = 1$, robust=0.1 case is nearly as good as the gridless uniform limit. For example, gridless weighting will nearly always help, but when compared against gridded weighting + robustness it might not help much. For individual cases, one needs to examine the sidelobes careful as well as the parameters shown here.

As discussed before, the snapshot case is unusual in that the super-uniformly weighted RMS is actually less than in the uniform case. Clearly from these traces, super-uniform weighting is a big win for snapshots. Notice that *both* traces of the gridless weighting lie essentially along the gridded super-uniformly weighted curve.

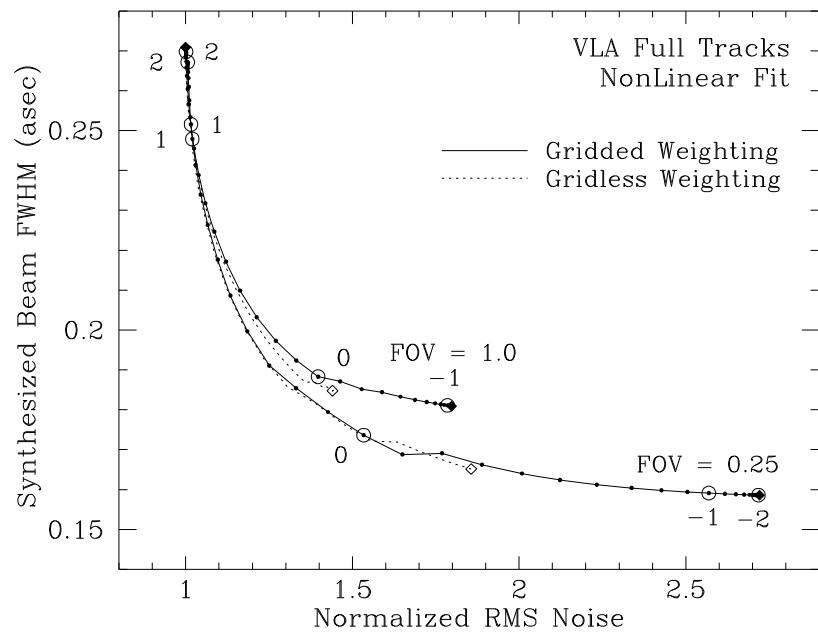


Figure 3.23a: Robustness tradeoff curves for the VLA Full Tracks case.

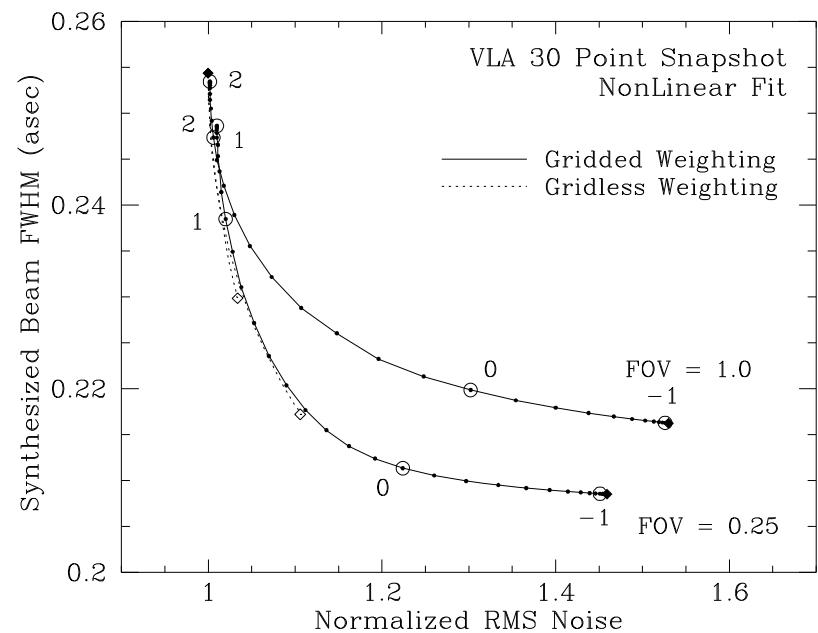


Figure 3.23b: Robustness tradeoff curves for the VLA 30 Point Snapshot case.

The robustness curves for the orbiter case are so dominated by the variations in individual snr weights that both super-uniform and/or gridless weighting make little difference. These techniques both affect the gridding irregularities, and that just isn't the problem here. The only parameter that makes significant difference is the robustness.

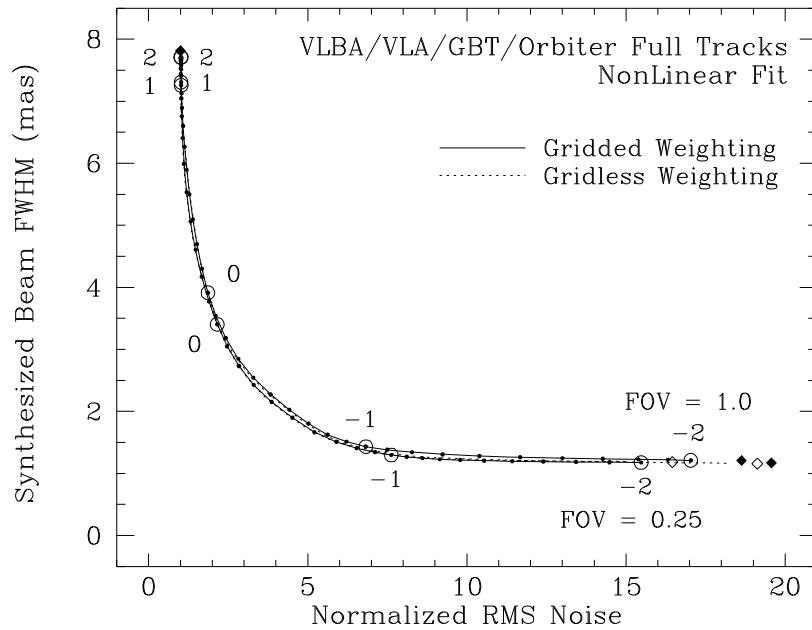


Figure 3.23c: Robustness tradeoff curves for the VLBA/VLA/GBT/Orbiter case.

3.8.2 Thresholding

It was shown in Section 3.6.3 that it can actually be better from a signal-to-noise standpoint to simply drop highly upweighted points than to include them. It may be counter intuitive that throwing away good data can *decrease* the thermal noise, but it is so. Here we have taken the Full Track VLA data and made an additional flagging pass after normal uniform weighting. A threshold was set which flagged bad all points where less than N visibilities were gridded into the same weighting box. Figure 3.24 plots the resulting parameters as a tradeoff curve against N . The percentage next to each point is the fraction of data which is discarded by the threshold. The number in parentheses is N . This is a very graphic demonstration that the uniform end of the tradeoff curve is driven be a small fraction of points that are amplified by the weighting. Throwing away these points acts like a crude version of robustness. Robustness is to be preferred, since the excised data is at otherwise unmeasured spatial frequencies, and we wish to preserve whatever information we can at those frequencies, provided that it not affect the RMS noise too strongly. But it is sobering how well this works near the uniform end of the curve. One can actually throw away 13% of the data and still decrease the RMS from 1.80 to 1.33! The thresholding curve diverges from the robustness curve when the value N approaches the typical occupancy in the outer tracks.

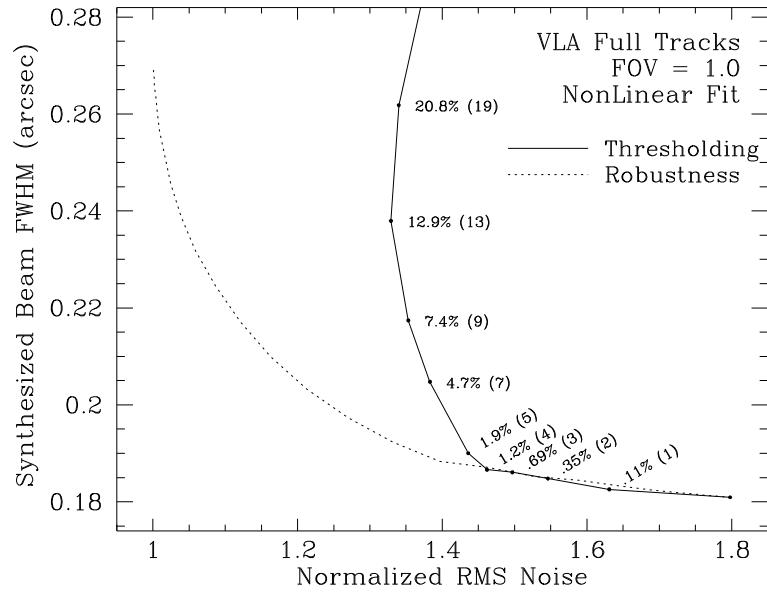


Figure 3.24: Thresholding tradeoff curve, VLA Full Track case. Visibilities less than a given threshold in gridded weight are simply eliminated from the data set. The remaining data are mapped with uniform weighting.

An even more extreme example of thresholding (not shown here) is from the VLBA/VLA/GBT data set. If one throws away *all* data from baselines between VLBA antennas, (presumably after self-calibration), one obtains a uniform beam superficially very

similar to the robust=0 case of Figure 3.21. Measured by RMS/resolution, it is really fairly close: 2.20/6.10 mas instead of the 2.06/6.10 mas produced by a robustness of -0.217 on the full set. The beam slices along the axes of the beam do not happen to pass through the worst sidelobes, and these too are strikingly similar to each other. The true behavior is revealed when we look at contour plots. While all the major features are present in both plots, both positive and negative sidelobes are much worse in the thresholded beam than in the robust beam. The worst negative is -14.8% compared to -8.1% . The robustness in this case has placed the uniform/natural threshold between the typical summed weights encountered on baselines between VLBA antennas and those encounter on baselines including a large element. The net effect is that all of the baselines with a large element are nearly uniformly weighted, and this is what is causing correspondence with the thresholded beam. But in the robust beam, all of the internal VLBA baselines are also present in the data, downweighted by roughly a factor of 10. The spatial frequencies present at this level are not enough to significantly change the beam features, but they help considerably to lower the beam RMS sidelobe level.

Thresholding works surprisingly well in terms of thermal RMS/resolution. Robustness is superior to thresholding because it preserves all the spatial frequencies in the data.

3.8.3 Time Averaging

As demonstrated in the snapshot simulations, time averaging can serve a similar function as does a moderate amount of robustness, at least in cases where the data is not intrinsically unbalanced by the snr weights. Examined in detail, there is a surprisingly rich variety of ways that uniform weighting and time averaging can interact to affect the relative weights at different spatial frequencies. But in practice, the dominant effects in most cases do much the same thing; an increased averaging time leads to more weight at the shorter spacings, a wider beam and lower thermal noise. Snapshots are an exception: a good choice of averaging time is quite useful, but an inappropriate choice can lead to dramatic degradation in sensitivity.

Recall that after uniform weighting, each nonzero term in the gridded Fourier sum contributes equally to the dirty beam. The resolution of the beam is set by the relative weightings of the different spatial frequencies, so what we want to know is the typical number of cells occupied at different u - v radii. The easiest case to understand is probably the densely sampled case, where there are a great many visibilities per cell towards the center of the sampling, and isolated tracks out towards the edge. As the averaging time is increased, there will be fewer and fewer occupied cells in the outer tracks. This is true even in the regime where there are many visibility points per average cell along those tracks, since the two-dimensional nature of the grid allows intersections of tracks and gridding cells of any length less than approximately the cell diagonal. Referring back to the catastrophic gridding error illustration, Figure 3.2, it is easy to imagine that averaging down by a factor of two could leave the cell with weight 1 unoccupied. The distribution of low-occupancy cells was discussed in Section 3.6.3 to explain the uniform limit of the robustness curve; the probability of finding a low occupancy cell rises dramatically towards the edge of the u - v sampling. The probability of an occupied cell becoming empty after time averaging is related to its original occupancy, so the outer tracks have a greater chance per cell of losing a term in the Fourier sum after time averaging. The result is a higher emphasis on the lower spacings and a wider beam. Note that the cells most likely to become empty intersect the very short tracks that nick the cell corners. Visibilities along these tracks were shown by the thresholding demonstration to be responsible for the uniform limit of the robustness curve. Time averaging is acting like a statistical form of thresholding, and once again we obtain the familiar behavior.

Three time-averaging tradeoff plots are given in Figure 3.25, covering the two VLA full track cases and the 30 point snapshot. The synthetic full track case resembles the uniform end of the robustness tradeoff curve for the reasons just given, though note that the magnitude of the averaging time required for a significant effect is quite large. Because of time smearing, we rarely wish to average beyond a few minutes. This only corresponds to a small amount of robustness, and time averaging is obviously not a terribly effective way to move around in the RMS-resolution plane. Still, if the tools for robust weighting are not available, the effect is worth remembering. A further reason to prefer robustness to

extreme time averaging becomes apparent when we examine a more realistic data set, with many short segments in the $u-v$ plane corresponding to individual scans. Time averaging has the potential to isolate $u-v$ points at the end of a scan in exactly the same way that points can be isolated in a reweighting cell. If the typical $u-v$ distance between averaged points is greater than the grid spacing, the same points which are isolated by time averaging will also be likely to fall in their own reweighting cell. The result is a new population of isolated highly-amplified weights which have the usual unpleasant effect on the thermal sensitivity. Figure 3.25b shows that this effect does not become important for a typical full track observation until well above the averaging times typically used. Figure 3.25c shows that this is not true for snap shots. Notice the difference between averaging 30 10-second time samples into 60-second time bins versus 70-second time bins. It is nearly a 20% degradation in sensitivity.

When time averaging snapshot observations, be sure to choose an averaging time that divides your data into roughly equal sized bins.

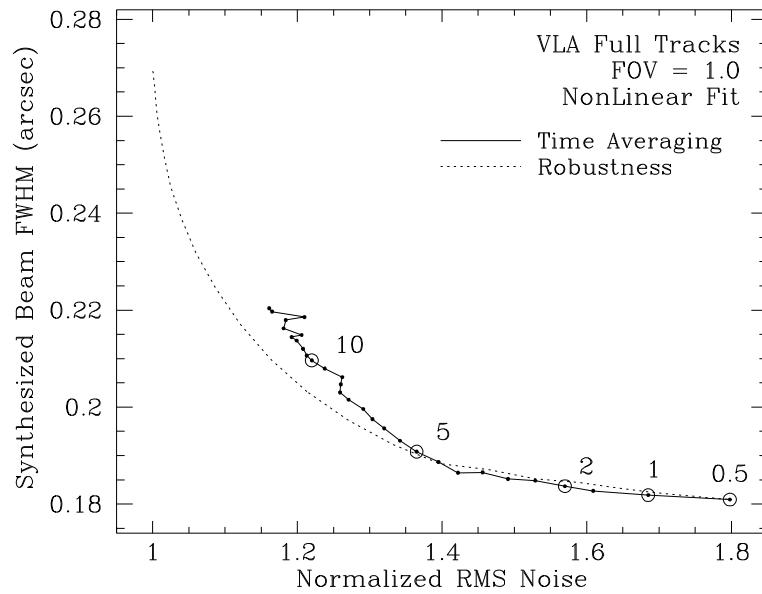


Figure 3.25a: Time-averaging RMS/resolution tradeoff, VLA Full Tracks. Numbers are the averaging time in minutes.

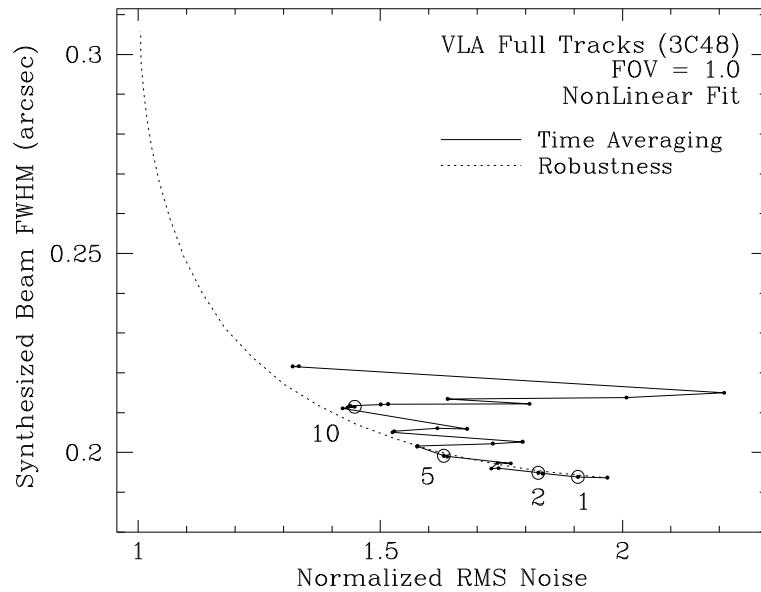


Figure 3.25b: Time-averaging RMS/resolution tradeoff, VLA Full Tracks (3C48). The large variations with averaging time are caused by points being isolated at the end of scans by the time averaging. Numbers are the averaging time in minutes.

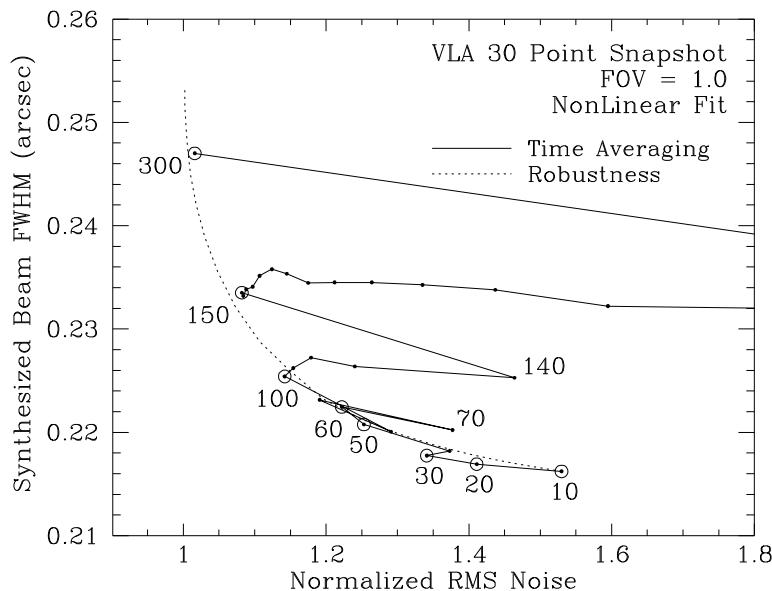


Figure 3.25c: Time-averaging RMS/resolution tradeoff, VLA 30 Point Snapshot. Numbers are the averaging time in seconds, (and also the number of points in each bin times ten). The circled points are those which divide the original points into equal sized bins.

3.8.4 Super-Uniform Weighting

Since all of the tradeoff curves go from a limit of natural weighting to a limit determined by the degree of super-uniform weighting, an obvious question is to ask what happens to that limit when the tradeoff parameter is held constant and the super-uniform parameter (weighting field of view) is varied. From the theoretical considerations in Section 3.3.4, it should come as no surprise that a smaller FOV_w leads to a sharper main lobe and poorer thermal noise. In fact, this effect is largely driving the RMS/resolution scatter plot against image parameters in Figure 3.1. For a more detailed interpretation, however, let us consider the effect of the reweighting region in the u - v plane.

In the sub-uniform limit ($FOV_w \rightarrow \infty$ or $\text{IMSIZ} \rightarrow \infty$), the size of the reweighting box goes to zero. If every visibility grids to its own u - v reweighting box, the density weights D_k are just the inverse of the snr weights, $1/w_k$. If the snr weights are all equal, the density weights are constant and we recover natural weighting. There are two complications in this limit. The first, obviously, is that the snr weights may not all be equal. The second is that some visibilities may have exactly the same u - v coordinates. No reweighting boxes can split the two visibilities, and again the natural weighting limit will never be reached.²⁰ The limit will be approached smoothly in resolution, as splitting one reweighting box into two merely results in one more term among many in the Fourier sum, the extra term preferentially appearing at the lower spatial frequencies where there are more points. The RMS noise will be dominated in the limit by the many points which grid into their own cell, and it will already be fairly close to one. The isolated point is now the norm rather than the exception. Splitting a cell with multiple points in it will drive the density weights closer to uniformity rather than producing a catastrophic reweighting, so the limit will also be approached smoothly in RMS.

The super-uniform limit ($FOV_w, \text{IMSIZ} \rightarrow 0$ or $\text{UVBOX} \rightarrow \infty$) comes when the weighting box goes to infinite radius in the u - v plane. In this limit, every visibility is considered in the weighting box of every other visibility and all are reweighted by the (constant) sum of the snr weights. We recover natural weighting no matter what the u - v coordinates or snr weights of the data. As to how it approaches this limit, however, we have to consider both forms of super-uniform weighting. The important thing is that the scaled cell super-uniform weighting will behave badly at with very large weighting boxes. Exactly the same catastrophic gridding error can happen as in Section 3.5, but in this case on a global scale.

The limiting cases are interesting, and of course the most tractable analytically, but in a practical sense we want to know what happens in the intermediate regions. Figure 3.26 gives several examples of actual RMS/resolution tradeoff curves parameterized by

²⁰If one were truly creative, a visibility set could be constructed such that these two effects exactly cancel each other — a data set containing two visibilities with the same u - v coordinates and snr weights each half that of the rest would preserve natural weighting in the sub-uniform limit.

FOV_w . The first panels are the VLA Full Track (3C48) case from before. Solid points are scaled box super-uniform weighting, and hollow points are the gridless weighting. Software limits prevent using the gridded uniform weighting algorithm in the extreme sub-uniform limit, and prevent using the gridless algorithm in the extreme super-uniform limit. The region of overlap is $1/4 \leq FOV_w \leq 4$. Notice that the gridless weighting curve always lies below and to the left of the gridded curve, which is generally an improved beam barring extreme changes in the sidelobes. The *AIPS* style integer box super-uniform weighting is not shown. Numbers next to the circled points are FOV_w . The light dotted lines are robustness tradeoff curves for the gridded $FOV_w = 1$ and $FOV_w = 0.25$ cases. The robustness curves do not quite reach $RMS = 1$ due to a minor inconsistency in how SDE weights the points along the $u = 0$ axis. The limits of Figure 3.26a are fixed at a reasonable value, but there are many points outside the plotting area. The worst RMS of the points examined came at $FOV_w = .0103$, and resulted in an RMS/resolution of $40.5/.144''$. The highest resolution came at $FOV_w = .00781$ for $16.0/.132''$. The connecting line between the gridded points has been suppressed for the uniformly-weighted cases because the points are jumping around so badly. Figure 3.26b shows the same data, but with a robustness of 0. Since much of the chaotic dependence on the high end tail of the weighting histogram is reduced by the robustness, we can now see the tradeoff curve approach the super-uniform limit comparatively smoothly. The sub-uniform limit is always approached smoothly by the gridless algorithm, when operating on the full track data. The uniform snapshot case is fairly irregular for both gridded and gridless cases, but it is clear that the weighting field of view is a very important parameters for snapshots. Some values of FOV_w produce quite favorable beam properties. Also, more noticeably here than elsewhere, gridless weighting is a particular improvement over gridded weighting.

Figure 3.27 shows beam slices and contours of the 3C48 case. Notice that the higher degree of super-uniform weighting leads to a tighter main lobe, as we have seen before, and also to deeper first sidelobes. Perhaps most interesting of all, notice how similar the sub-uniformly weighted $FOV_w = 4$ case is to the Robust = 0.5 case of Figure 3.13.

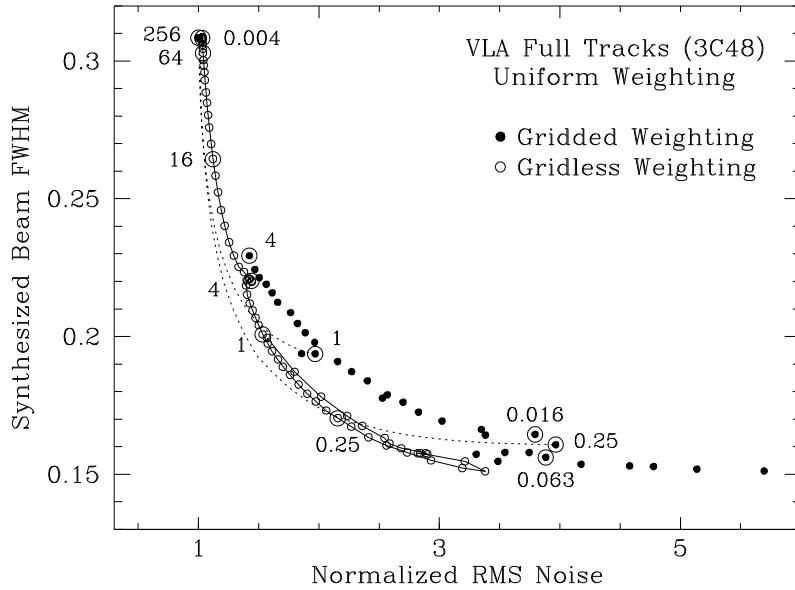


Figure 3.26a: RMS vs. Beamsize as FOV_w is varied, VLA Full Tracks case and uniform weighting. Circled points are labeled with FOV_w , and the light dotted lines are robustness tradeoff curves for comparison. See the text for full details.

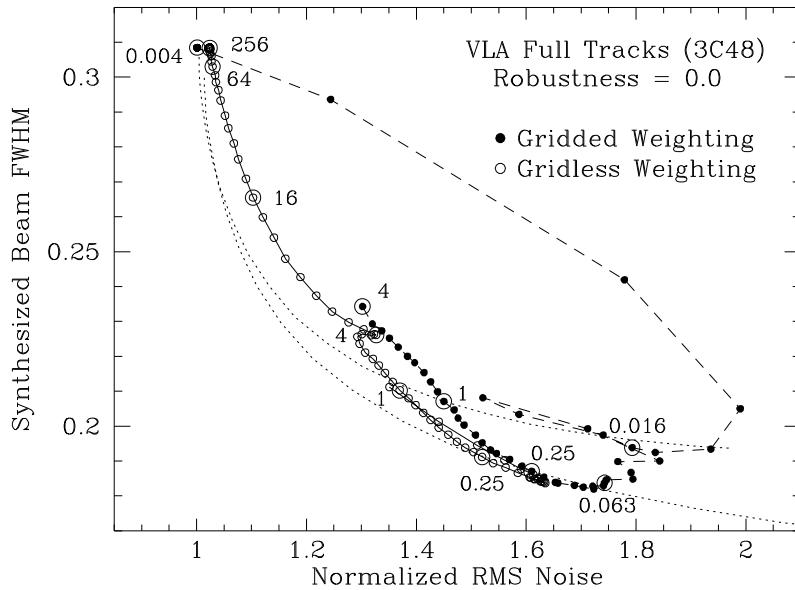


Figure 3.26b: Same figure as before, but with a mild value of robustness instead of uniform weighting. The dependence on FOV_w is now smooth enough that the gridded points can be connected.

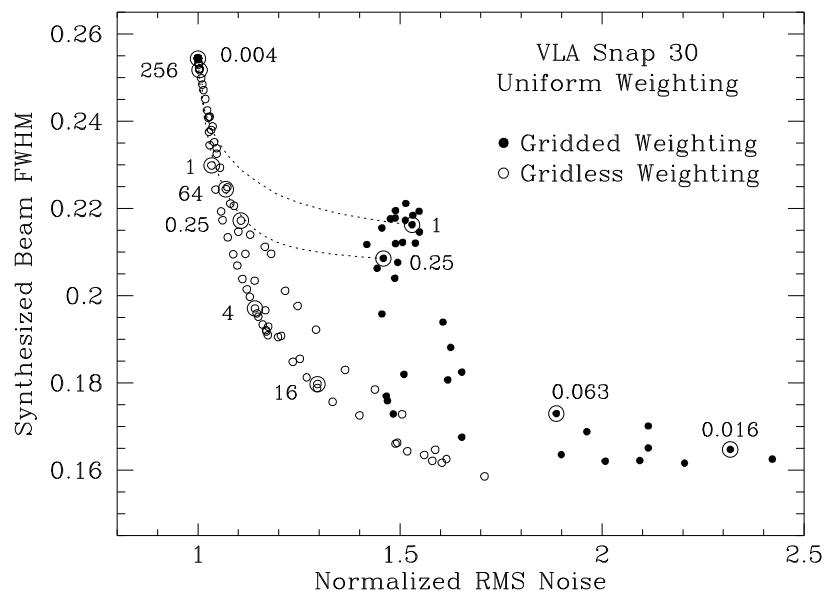


Figure 3.26c: RMS vs. Beamsize as FOV_w is varied, VLA 30 Point Snapshot case and uniform weighting. The small number of points in a snapshot lead to irregular behavior, but some values of FOV_w are clearly quite favorable.

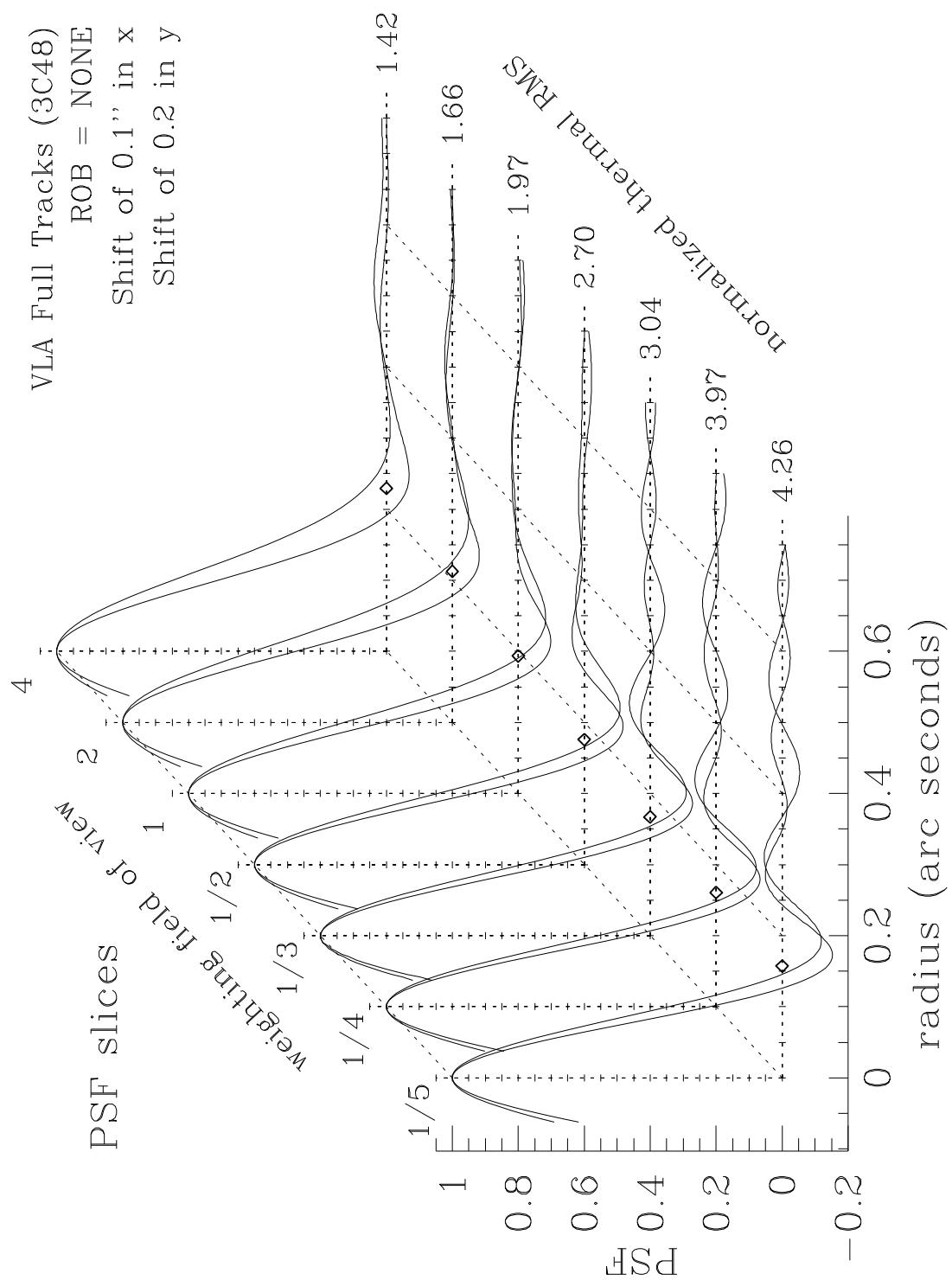


Figure 3.27a: PSF plots against FOV, VLA Full Tracks (3C48)

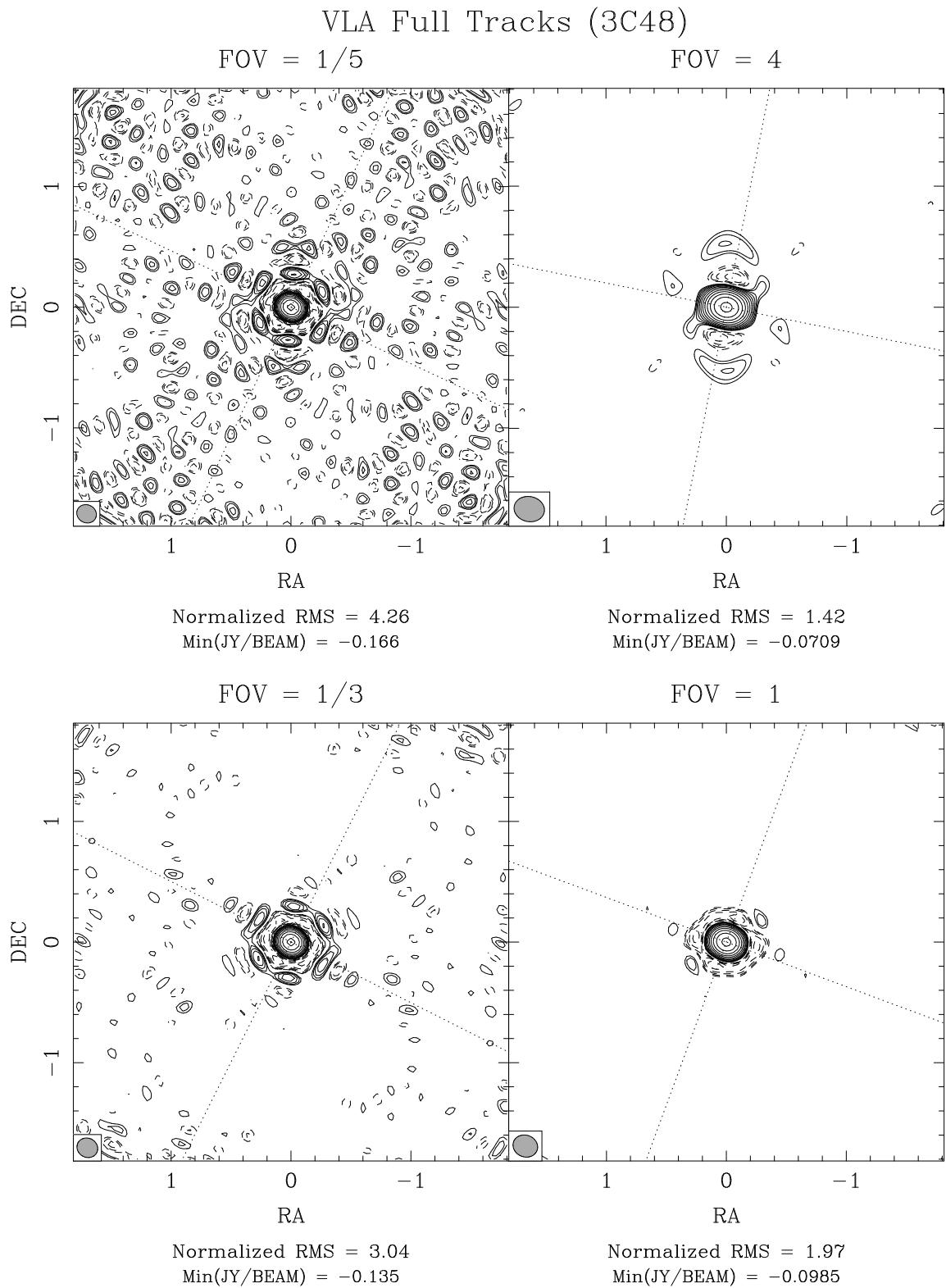


Figure 3.27b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

Occasionally there may be use for extremely large $u-v$ gridding boxes with sparse coverages. In this case, one is smoothing over not only the local tracks, but also over adjacent tracks and even over gross features such as the arms of the star pattern. In such cases one then usually tapers the outer spacings back down to avoid destroying the sensitivity. Since one is fighting two conflicting parameters against each other and operating in the region of compromise, it is difficult to give general guidelines of exactly when this will be useful or how much of a given parameter to use. In some cases, however, it can be useful to control the far out sidelobes of the beam. This should be examined on a case by case basis, but it is likely to be most useful for imaging crowded fields with marginal coverage.

Finally, we show an example from the VLA Multiple Configuration (M82) case. In this project, a number of images at different resolutions were required to create spectral index maps. The uniform and naturally weighted resolutions were $.085''$ and $.353''$. A map at a resolution of $.35''$ was desired, and the uniform beam was forced to this size with the tapering routines of Section 3.11. The image was sampled at $.35''/4 = .0875''$ and the image size was 2048^2 , which means that the effective degree of super-uniform weighting is a factor of 8.75 less than simulations in sections 3.7 and 3.11. The fitted beamwidth for the natural beam is a strong function of cellsize in this regime, which explains the difference from the $.285''$ reported in the earlier section. The tapered uniform beam is the left panel of Figure 3.28. This is a very nice beam, as might be expected from such a good data set. The significant taper is softening the effect of the density weights on the RMS; the resulting RMS of 1.86 is a considerable improvement on the untapered uniform weighting's 3.11. When confronted with such a beam, one might well be tempted to stop. There is still a significant extended shelf on the beam, however. Since an increasing degree of super-uniform weighting leads to deeper negative sidelobes, an obvious approach is to super-uniformly weight in the hopes of cancelling the shelf with a near-in negative. As shown in the right panel, a FOV_w of 0.2 leads to a considerable improvement in beam shape *and* also in RMS. In all respects, the super-uniformly weighted beam is an improvement on the uniform beam. Super uniform weighting can often be used as a final tuning adjustment when the gross properties of the beam are being controlled by another parameter. After one has found a local optimum in robustness or tapering, adjust the weighting field of view and reoptimize. As just shown, the results can be surprising and profitable.

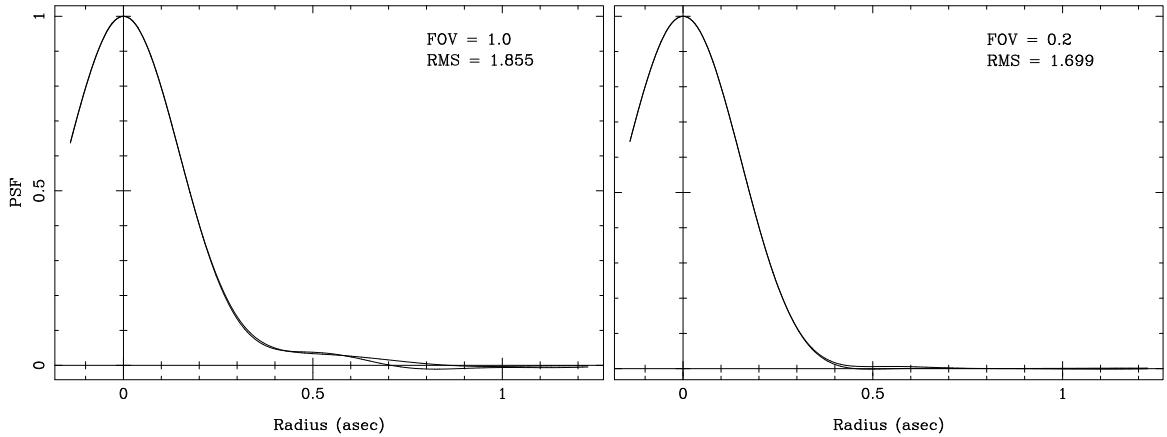


Figure 3.28: Super-uniform weighting as a way to combat an extended shelf. Both beams have been forced to a width of $.35''$.

Summarizing the properties of super-uniform & sub-uniform weighting:

- Sub-uniform limit ($FOV_w \rightarrow \infty$ or $\text{IMSIZ} \rightarrow \infty$)
 - Limiting case is (usually) natural weighting
 - Approaches limit smoothly
 - Not natural weighting if snr weights not equal
 - Not natural weighting if some points have identical coordinates
- Super-uniform limit ($FOV_w, \text{IMSIZ} \rightarrow 0$ or $\text{UVBOX} \rightarrow \infty$)
 - Limiting case is always natural weighting
 - With scaled cell super-uniform weighting, will approach the limit quite abruptly.
 - With integer box or gridless super-uniform weighting, will approach the limit smoothly.

3.9 Automatic selection of Robustness

The ‘compromise’ beam roughly intermediate in resolution between the uniform and natural limits can have very favorable properties in terms of close in sidelobes for some samplings, as demonstrated in the VLA Full Track and other simulations. It’s not terribly deep why this occurs. Since the VLA among other arrays has an excess of short spacings, the naturally weighted beam has a strong extended shelf. The uniformly weighted beams invariably have a strong first negative sidelobe, due to the sharp edge at the edge of the sampling. Since the robust beams vary between these extremes, there will be some value of the robustness which roughly zeros the first sidelobe. If the particular sampling is suitable for this technique, a single value of the robustness may roughly zero the worst of the sidelobes at all position angles. If the sampling is intrinsically very azimuthally dependent, a generalized extension of robustness might help — but preliminary results presented in Section 3.12 show this not to be terribly straight forward. For the moment, such samplings are simply beyond the range of the technique. The other useful properties of robustness remain, but near-in sidelobe control via robustness isn’t very effective.

Since the shelf or the first negative is the dominant mismatch between the beam and its own fitted Gaussian, we can use the integrated mismatch as a figure of merit and vary the robustness to minimize it. The important fact to take from this section is that the particular choice of error metric is not terribly important. The error curves given in Figure 3.29 are quite typical, and are not very sensitive to the integration region. In the case shown, VLA Full Track (3C48), all three error metrics are suggesting a value near 0.5 as the best value. Varying the region of integration from a circle with radius 1.5 times the fitted half width to 8 times the fitted halfwidth, (the fitted Gaussian varies from .21 to 5×10^{-20} over the same range), the robustness selected varies at most from extremes of .4 to .7, with most estimates being between .5 to .6. In fact, all of these values of the robustness are fairly reasonable choices. A subjective choice of between .55 and .6 as the optimum value was made after examining all the contour plots by eye, so the automatically selected and human selected values agree quite well.

The approach of selecting a robustness by minimizing the misfit to the fitted Gaussian is implemented as the SDE shellscript `findrobust`. It uses Brent’s zero finder as described in Press et al. (p. 283, 1986), and simply begins with an initial guess of (3, .5, -3) for values of robustness that bracket the minimum. In all cases examined, the initial guess was adequate and the convergence quick; typically 8 executions of the mapping program was sufficient to constrain the robustness to within ± 0.01 . A human being can usually home in on a reasonable value of robustness for a Gaussian-like beam by testing a few cases, but this is a simple enough procedure that is worth implementing for the convenience value alone.

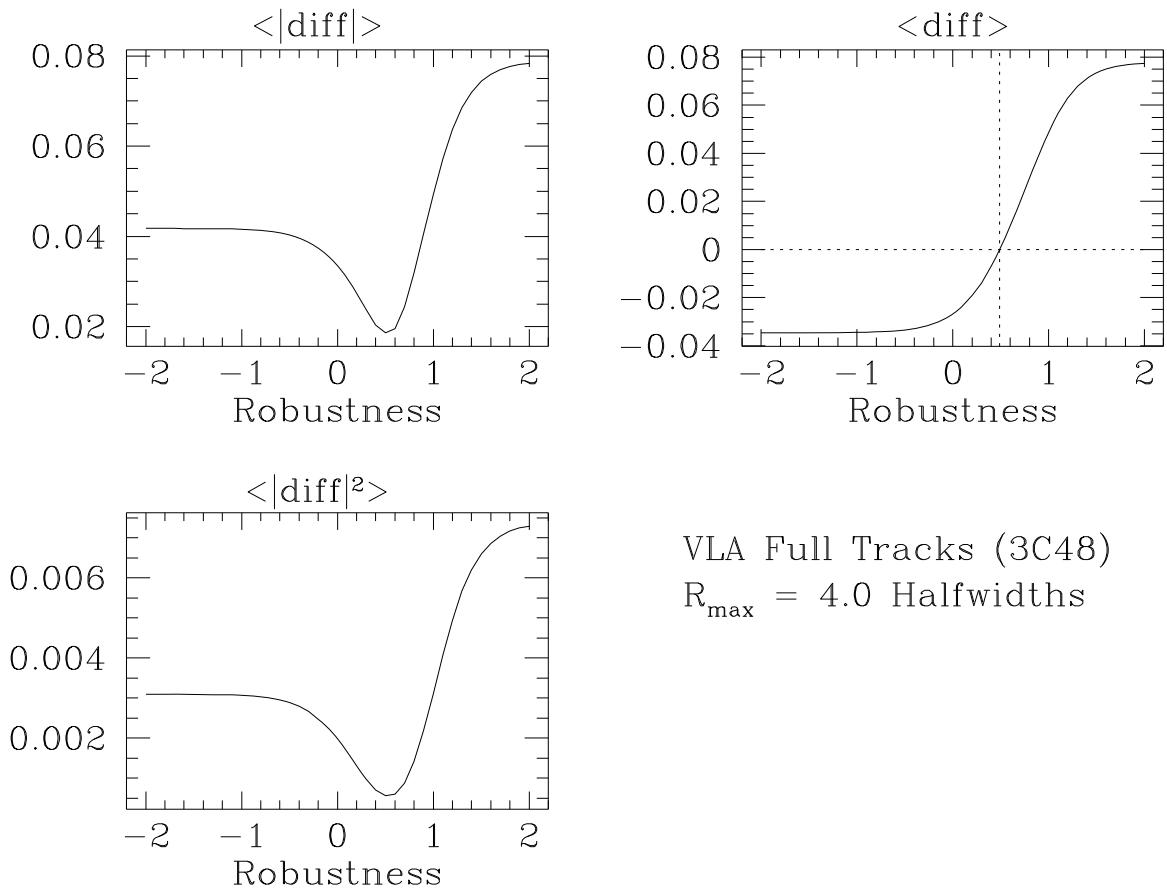


Figure 3.29: Automatic selection of robustness. The curves are possible misfit criteria between the main lobe of the beam and its fitted Gaussian. All here are indicating a value of Robust = 0.5 as the optimum.

3.10 Model Deconvolutions

The end result of the weighting process is a dirty map, which will then be deconvolved. It is the errors in this deconvolved map which concern us, of course. The subject of deconvolution errors in general is complicated, and comprises much of the remainder of this dissertation. Here we simply present one particular deconvolution of a model source as a function of robustness. The model source is one we will see several times in later chapters. It is a 4-component Gaussian model derived from a fit to the X-band images of 3C48 in Chapter 7. Additionally, 4 point sources have been placed around the main source to show the effect of the thermal noise. The height of the weakest point source was chosen to be three times the naturally weighted thermal noise. Each additional point source was increased in magnitude by one contour level, the fourth root of ten ≈ 1.78 . The u - v coverage is that of the synthetic VLA Full Tracks case.

Flux (Jy)	RA offset (")	Dec offset (")	B_{maj} (")	B_{min} (")	B_{pa} ($^{\circ}$)	Type
1.9059	.0138	.0513	.13151	.08392	178.77	GAUSS
1.0775	-.0615	-.1617	.09857	.02470	168.04	GAUSS
.4798	.1686	.1119	.46544	.33404	10.31	GAUSS
.0286	-.1845	-.5601	.32188	.14573	12.40	GAUSS
.0048	-1.2	1.2	—	—	—	POINT
.0085	-1.2	-1.2	—	—	—	POINT
.0152	1.2	-1.2	—	—	—	POINT
.0270	1.2	1.2	—	—	—	POINT

Table 3.8: 3C48 & point source model parameters

This is actually a fairly challenging source at high fidelity levels, and the NNLS deconvolution algorithm described in Chapter 4 does a considerably better job than CLEAN on the noiseless data. For the thermally-dominated data presented here, nearly any reasonable deconvolution algorithm will give roughly the same result. Error plots are given of the maximum error over the entire image, the maximum error off the support of the source, and the RMS error over the support of the source. These three quantities are selected to show the worst-case error, the worst error that an astronomer might notice on a map, and the off-source noise.

These simulations were done in the u - v plane — the visibility data was simulated, rather than forming the dirty map with a discrete convolution. The model shown in Table 3.8 has some serious representation problems, even when oversampled at $\Delta\ell = \Delta m = 0.03''$ as here. The second Gaussian in particular has a minor axis width about the size of a pixel. Consequently, it makes a considerable difference how one generates the truth image for the error plots. As mentioned in Section 2.6, the right way to do it is to smooth the model analytically or in the visibility plane, and then sample. Sampling the model and then smoothing results in an error of more than 5% at the location of the second Gaussian.

This is large enough to dominate the full error plots, even the noisy ones. If the comparison image is made properly, representation problems have almost no effect on the error plots.

As demonstrated in Chapter 5, the dominant error in CLEAN reconstructions occurs immediately outside the envelope of the u - v sampling. The size of the restoring beam determines how much of this spectral error is included in the smoothed components. Thus if one uses a restoring beam fitted to the main lobe of the dirty beam, as is the usual practice, the naturally weighted maps have an advantage in that the wider restoring beam includes less of the deconvolved model from the unsampled regions of the u - v plane. The error curves here all include two traces: one using a fixed restoring beam which is indicative of how well the deconvolver has performed in an absolute sense, and another using the fitted restoring beam which is what the astronomer will see when following usual practice.

Deconvolution simulations are difficult because there is an enormous parameter space to be explored. The worst problem is that solutions vary enormously with model — any given case should be simulated with an appropriate model, and generic results regarded with caution. But even apart from that, decisions about control parameters can also make a considerable difference in the qualitative nature of the results. Here, the question was whether to CLEAN the noiseless cases to extreme numbers of iterations, or simply to a “reasonable” level. A compromise was selected, and all cases were CLEANed to 20,000 components at 0.1 loop gain. A medium-tight support window of 4500 pixels was used, produced by smoothing the model with a circular Gaussian of width $.2214''$, and allowing flux wherever the smoothed model exceeded 10^{-5} of its peak. These images are CLEANed deeper than most people would choose, but the noiseless cases are somewhat under CLEANed for the highest possible fidelity. In the noiseless simulations, the maximum residuals rose from a minimum of $13 \mu\text{Jy}/\text{beam}$ for uniform weighting to $83 \mu\text{Jy}/\text{beam}$ for natural weighting. The trend reverses in the noisy case, with the maximum residual being $360 \mu\text{Jy}/\text{beam}$ for uniform weighting and $190 \mu\text{Jy}/\text{beam}$ for natural weighting. These numbers can be compared with the RMS off-source errors in Figure 3.31c. The RMS degradation/mean beamwidth for the weightings used are $2.03/.171''$, $1.44/.183''$, $1.09/.221''$, and $1.00/.271''$, for Uniform, Robust = 0, Robust = 0.6, and Natural weighting respectively. For the noisy case, .919 Jy of pseudo-random Gaussian noise was added to each real correlator channel in each visibility. The expected naturally weighted image-plane thermal noise is $1.59 \text{ mJy}/\text{beam}$.

The CLEAN deconvolutions are given in Figure 3.30a, and the model source smoothed to the corresponding fitted beam is given in Figure 3.30b. For the cases dominated by thermal noise, the smoothed model can be considered the expected improvement in the image; remember that the deconvolutions given here are only one realization of the thermal noise. In the plots, the lowest contour is twice the measured off-source RMS for each individual panel. Hence the apparent noise value stays roughly the same as the weighting varies. The sources rise up out of the noise floor as the weighting improves the sensitivity. For contrast, the similar sequence of plots in Chapter 7 is done with fixed contour levels, so the variation of noise with weighting is even more apparent there.

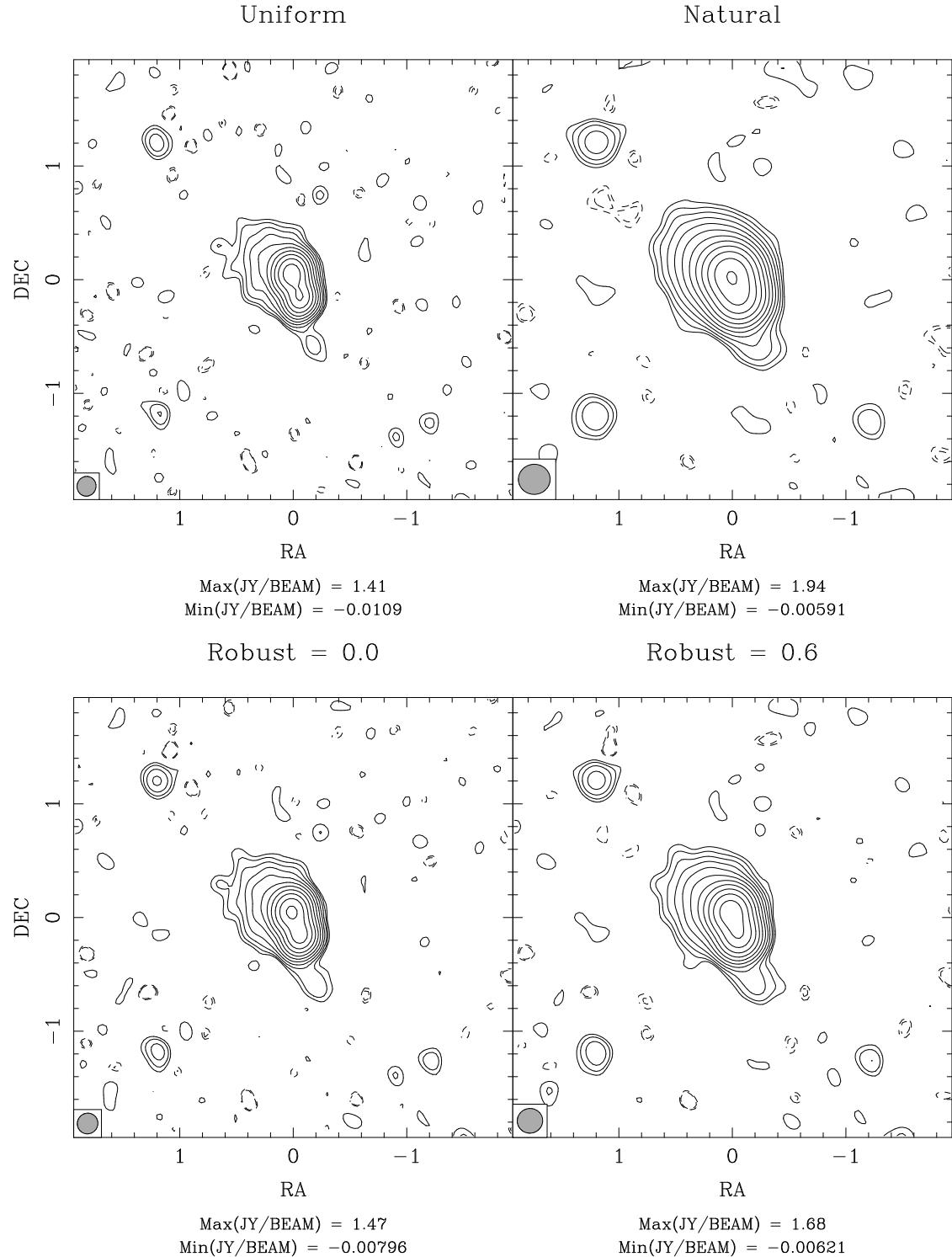


Figure 3.30a: 3C48 model deconvolutions. 4 logarithmic contours per decade. The lowest contour is 2 times the measured off-source RMS in each panel, so the source rises out of a constant noise floor.

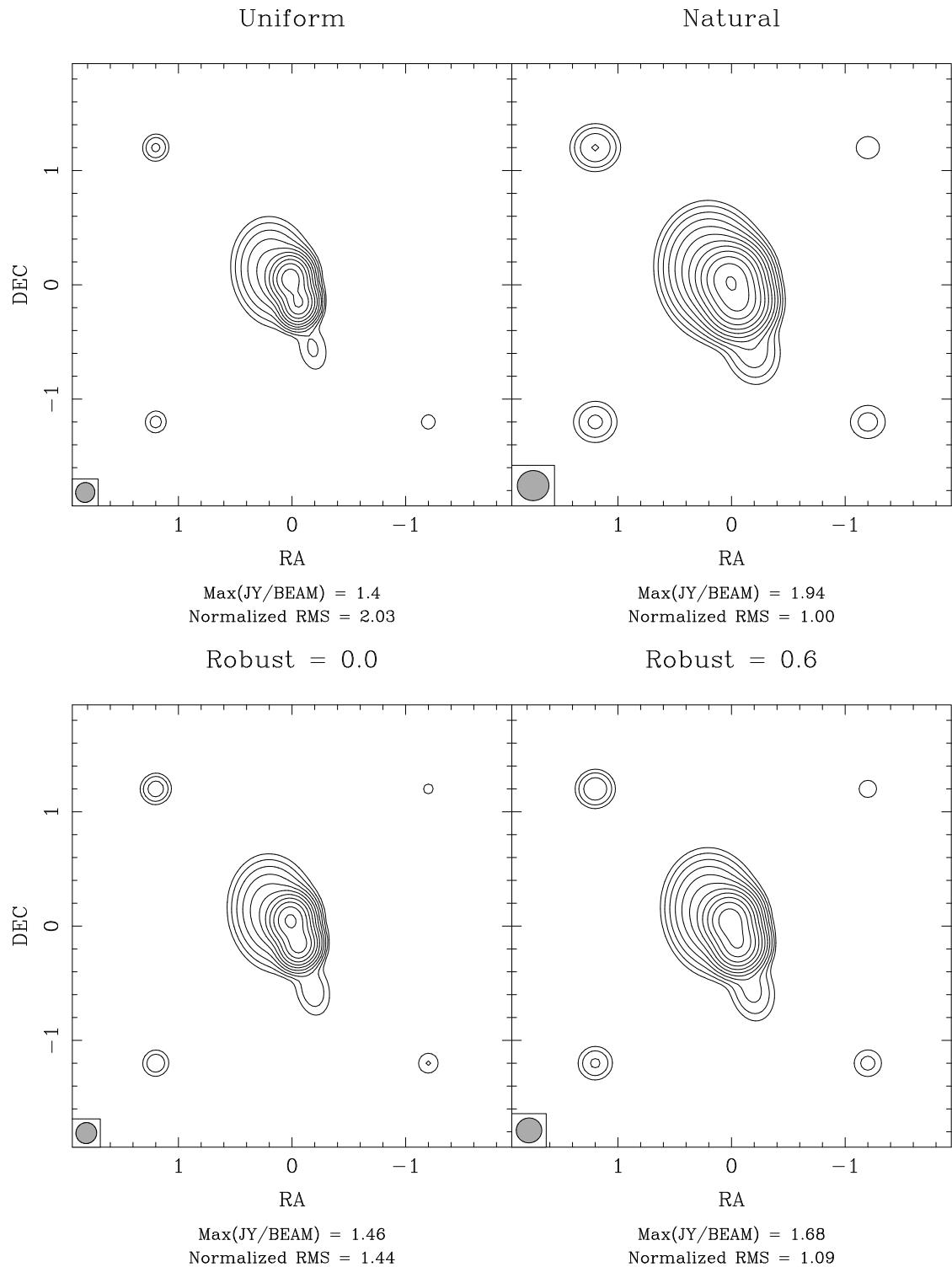


Figure 3.30b: Model contours to same levels as Figure 3.30a. Normalized RMS noise is given in each panel label.

We now examine some quantitative error measures as a function of weighting. Similar plots to these for the algorithms NNLS and MEM are later given in Figures 4.9 and 5.21. Some general comments and points to note about the error plots are:

- Robustness generally makes a difference of roughly a factor of 2 or 3 in the various error quantities. In Chapter 5, it will be shown that source structure makes a difference measured in orders of magnitude.
- The deconvolution behavior with weighting is complicated in the noiseless case, and varies greatly with algorithm.
- The maximum error plots tend to have the most structure with weighting, followed by the maximum off-source error. Off-source RMS with weighting is usually fairly simple.
- The worst deconvolution errors here are similar for the noisy and noiseless cases — even in a fairly noisy image, the worst on-source errors are coming from deconvolution errors and are not thermally driven.
- The maximum off-source error is significantly different for the noiseless/noisy cases. The noisy curve has features attributable to both the noise and to deconvolution errors.
- The noiseless maximum off-source error curve peaks at an intermediate value of robustness, as does the RMS off-source error. This behavior varies dramatically with deconvolver. In the CLEAN case, it's not likely to be a problem, since the cases where the effect is noticeable are where one will likely be using a different deconvolver.
- The measured off-source RMS after deconvolution tracks the predicted thermal RMS almost exactly for the noisy case. *The predicted thermal noise is a good indicator of what one is likely to measure in an actual deconvolution!*
- The uniform beam is a good choice for off-source RMS in the deconvolution error dominated regime, though this may simply reflect that the uniform beam tends to require fewer iterations to converge to a solution than the natural beam. This regime is somewhat under CLEANed in these simulations, and the residuals are nonnegligible.

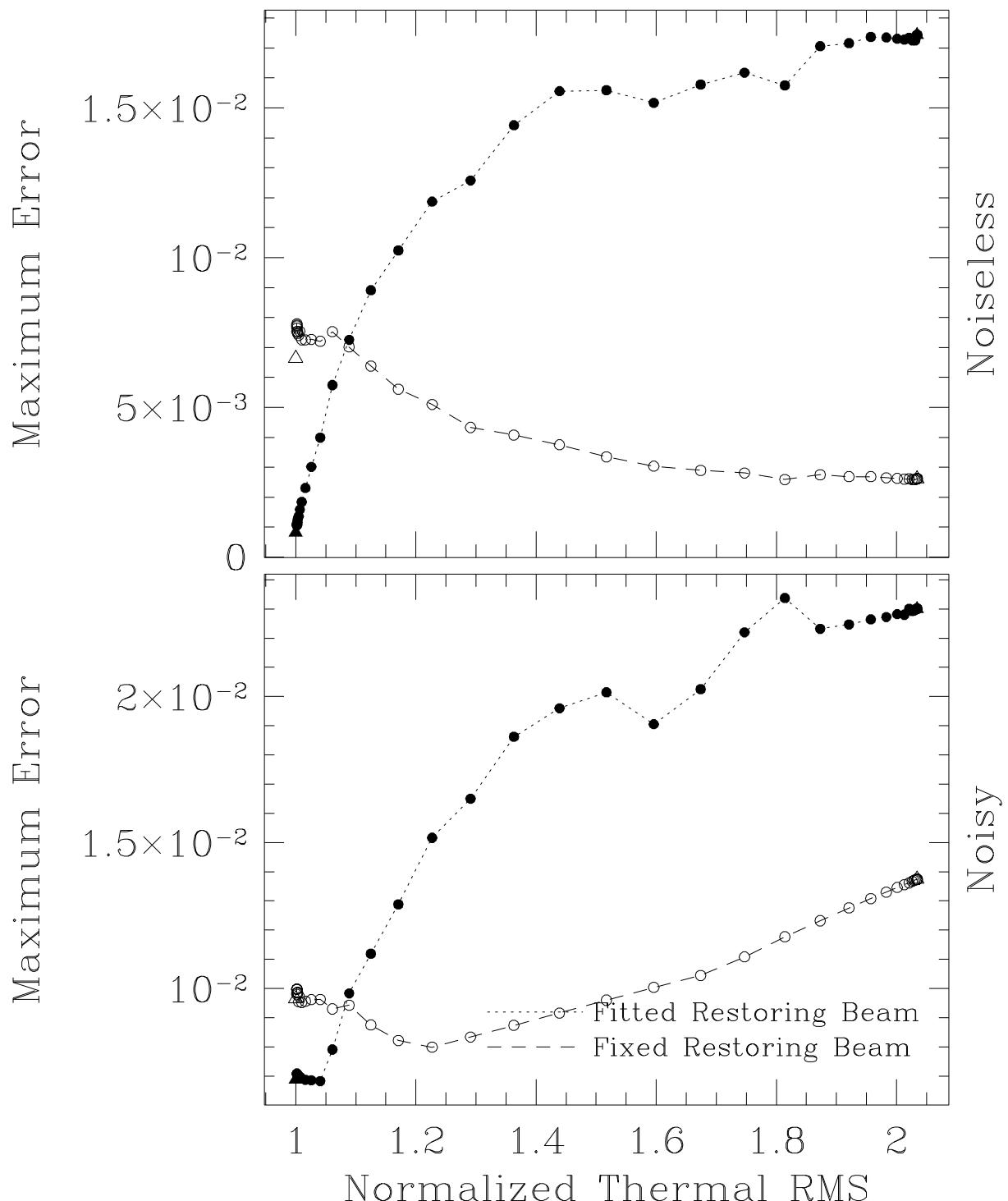


Figure 3.31a: Error quantities against RMS degradation factor: maximum error. The fitted restoring beam corresponds to Figure 3.30. The fixed beam curve is more representative of the absolute deconvolution quality and used $\sqrt{B_{nat}B_{uni}} = .2214''$.

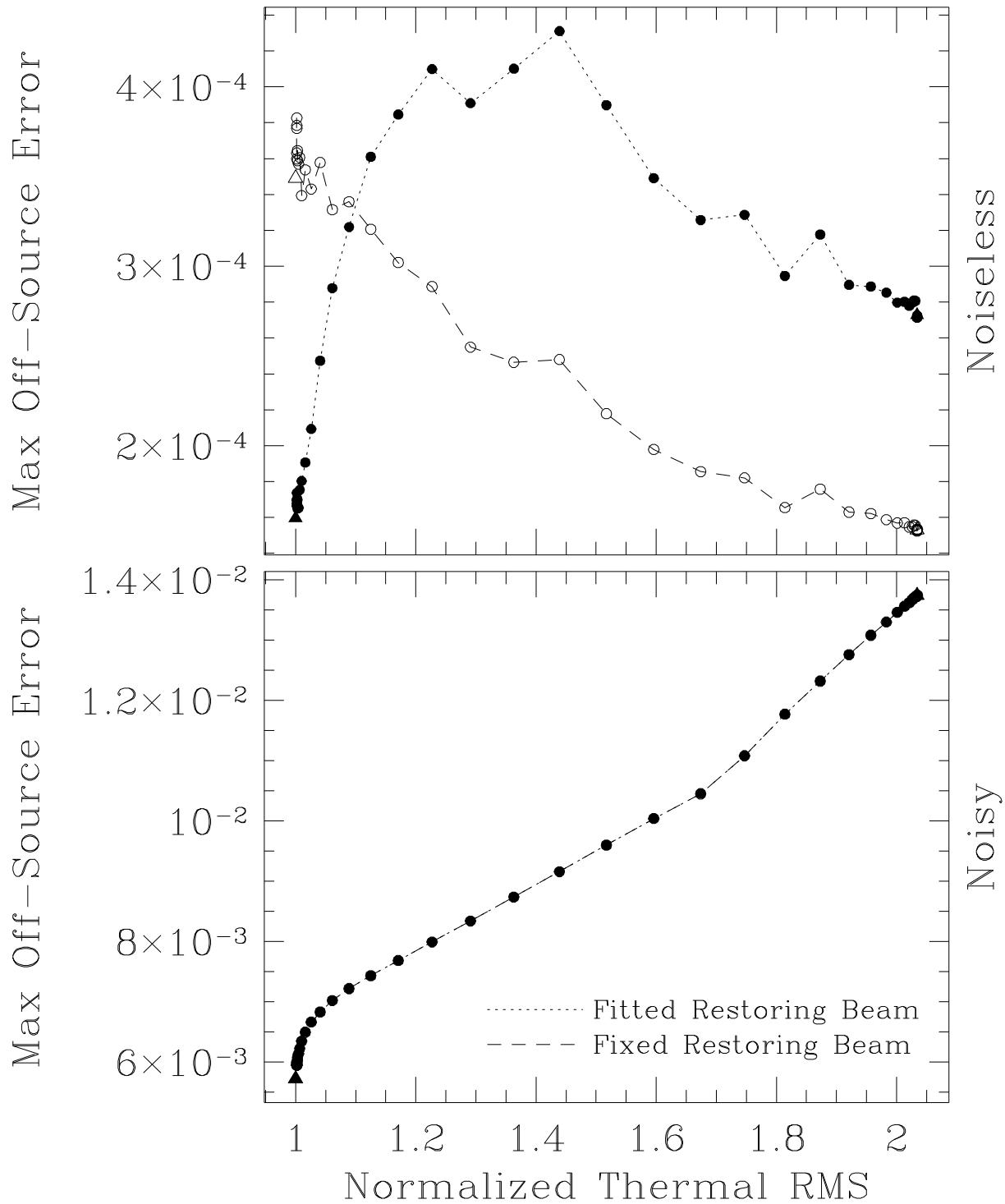


Figure 3.31b: Error quantities against RMS degradation factor: maximum off-source error.

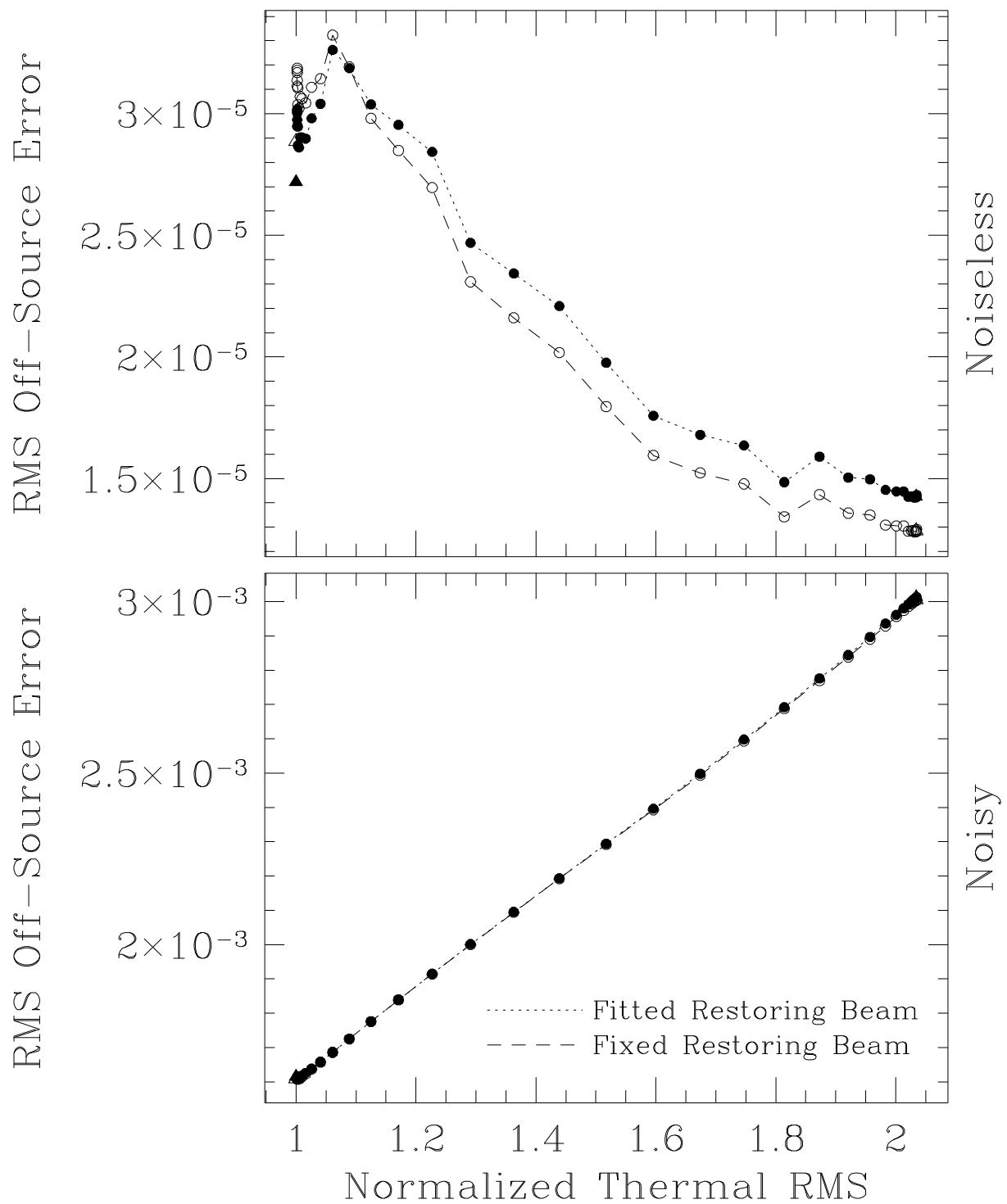


Figure 3.31c: Error quantities against RMS degradation factor: off-source RMS error.

3.11 Tapering & Beam Forcing

Tapering is a method of beam control that I feel is unjustly neglected. Most packages allow at least a one-dimensional radial taper on the visibility data applied after the density weighting. *AIPS* allows a different taper to be applied along the u and v axes. But few packages allow for a taper in full generality. Appendix B describes the necessary mathematics for a full two-dimensional rotated elliptical Gaussian taper. Appendix D describes how to generalize the taper to elliptical inverse tapers, or even to a hyperbolic Gaussian taper which one tapers along one axis and inverse tapers along the other. (A curve of constant taper is then a hyperbola in the $u-v$ plane.) The reason that one might want such a general taper is that the solution to the deconvolution of one elliptical gaussian from another involves such terms. With this formalism, we can work out what generalized taper we must apply to the data in order to achieve a given beamshape. We approximate this by analytically deconvolving the desired beamshape from the fitted beam. The resulting generalized taper is then applied to the visibility data and iterated. With some caveats, this technique works quite well. The details of such beam forcing are also given in Appendix D.

The first and most obvious use of beam forcing is in making spectral index maps. Within reasonable limits, this allows one to match any two fitted dirty beams to each other exactly. This is not explored in detail in this dissertation, but it should be quite obvious that a good match of fitted beams at different frequencies will lead to good spectral index maps.

A more mundane use of beam forcing is simply to circularize the beams for aesthetic reasons, or to minimize the effect of the beam position angle on source structure. This is not as effective as one might think for low declination sources for the simple reason that one cannot inverse taper data that is not there. But circularization by tapering works well with relatively dense samplings, and it can reduce the aspect ratio of extremely elongated beams at least somewhat. It is natural to worry about inverse tapering to a level that interferes with the deconvolution. In the limit of extremely high taper, clearly one is trying to image with just the outer $u-v$ tracks in the data. This is really no different than the super-uniform weighting case with very small FOV_w . As with that case, the normalized RMS is a reasonable guide for “how much is too much inverse tapering?” Also, note that the high taper limit of the beam in a case with circularly symmetric sampling is $J_0(r)$, the Bessel function of the first kind with order 0. This function has negative sidelobes of 40%, so clearly excess inverse tapering will be reflected in unreasonable high negative sidelobes. Figure 3.32 shows two examples from the AT Full Track (SN1987A) case used before. Since the $u-v$ coverage from the east-west array lacks a pie sliced shaped region, the beams from this coverage are moderately eccentric. As shown in Figure 3.6, the degree of eccentricity increases with robustness, naturally weighting generally being the worst, so here we only examine the Robust=0.5 and Naturally weighted cases. In the former case, the beam parameters are $1.11/1.096'' \times .856''$. After running `findtaper`, a beam of $1.16/.969'' \times .968''$ results. For the natural case, the original $1.00/1.324'' \times .967''$ becomes $1.05/1.132'' \times 1.131''$.

While the naturally weighted beam is not particularly attractive, circularized or not, the effect on the main lobe is quite clear and the RMS has not increased particularly much. The robust weighted case seems to be an improvement in most respects over its noncircularized counterpart, possibly excepting extremely high resolution projects.

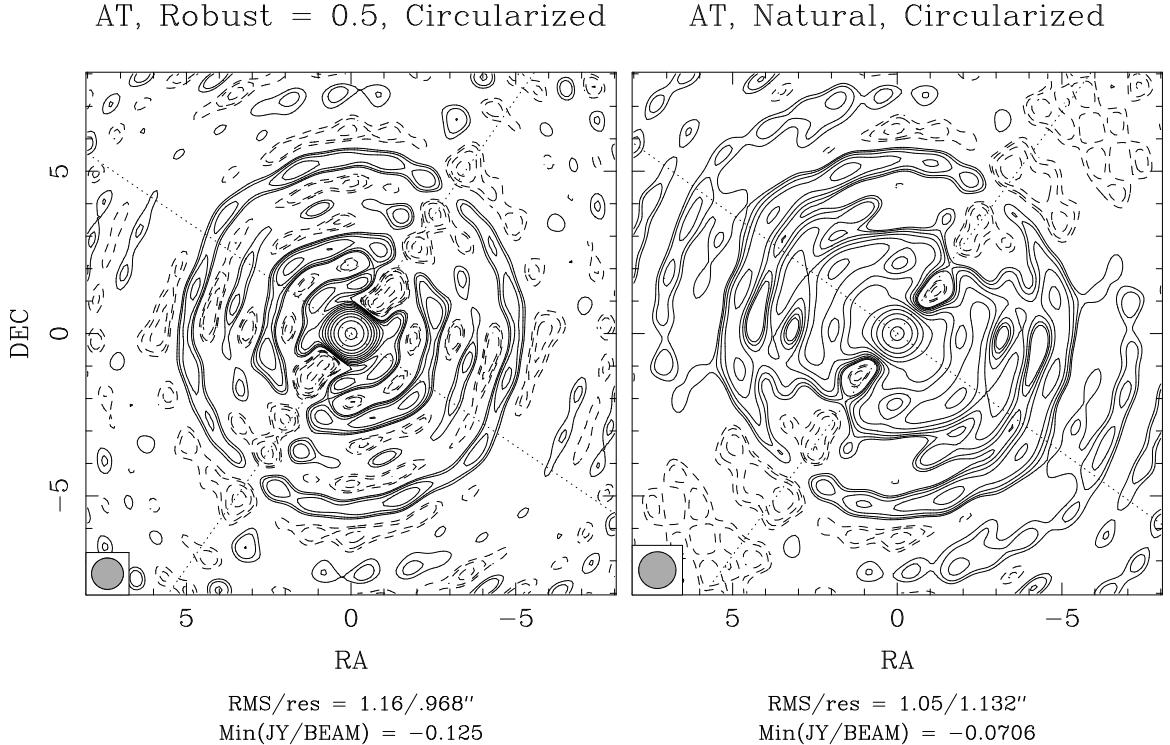


Figure 3.32: Beam circularization by tapering

We next examine the effect of simple tapering on a data set. Again we return to the VLA Multiple Configuration (M82) case. In Figure 3.33, beam slices are presented after the has been tapered down to multiples of $.15''$. This was done with beam forcing, but for most of the larger width curves it could also have been done with conventional packages in that the required taper was nearly a function of radius only. To avoid undersampling the higher resolution beams, these slices were done with the imaging parameters of Table 3.7 and not those of Figure 3.28. (The effective degree of super-uniform weighting is quite different in these cases.) The points to notice here are that the beam shape approaches a Gaussian more closely as the taper is increased, that there is a minimum in thermal noise as a function of taper, and that there are significant differences in beam shape as a function of FOV_w , even after significant tapering. None of these points are particularly surprising, but it is instructive to example the data from an actual observation.

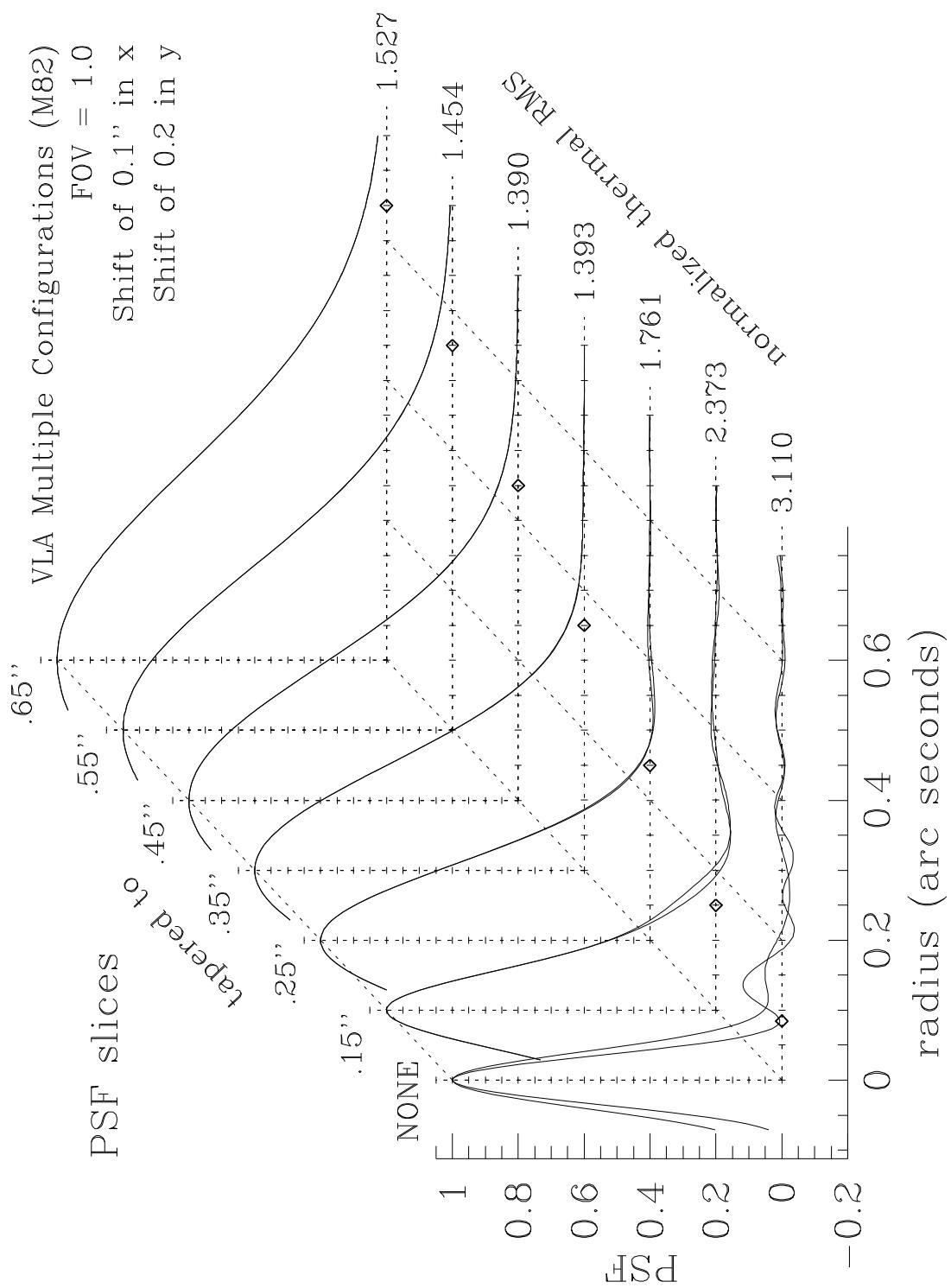


Figure 3.33a: Beam slices against tapering radius (arc seconds)

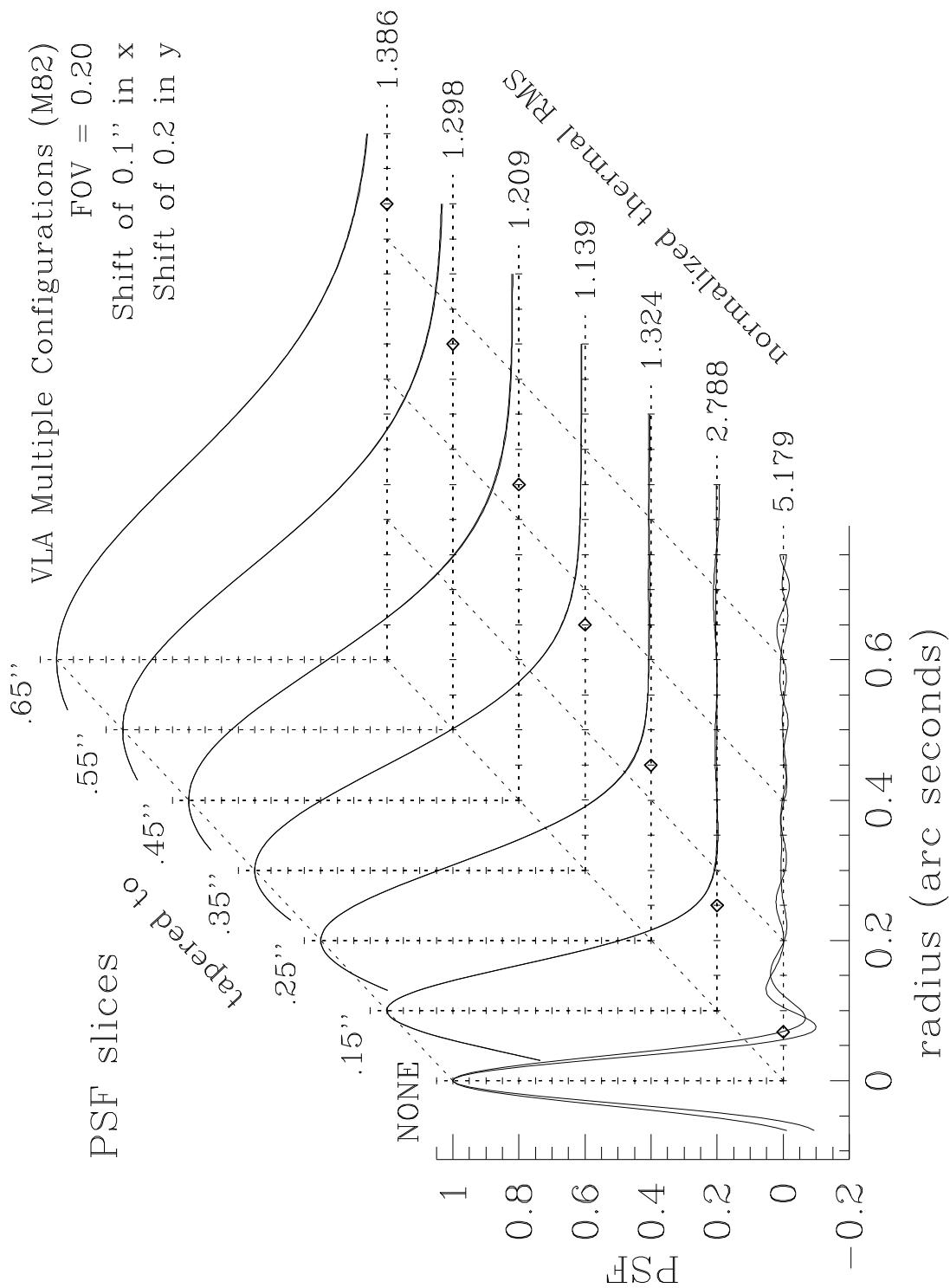


Figure 3.33b: Same as before, but with $FOV = 0.20$

The other particularly useful application of beam forcing is that it allows us to answer questions like “how much better is one weighting than another, after accounting for differences due to beam shape?” We simply match the beams as best possible using the beam parameter under study, and then force it the rest of the way with a generalized taper. Alternatively, we can match one beam to a second with taper, and then match the second to the first with a different taper. If one beam is superior to the other in both cases, we can claim it is superior without qualification as to beam shape. So armed, we tackle the question of gridded and gridless weighting again. Table 3.9 show this technique applied to a number of different cases of $u-v$ coverage, weighting field of view and robustness. The middle columns show the RMS of the beams when the gridless beams are forced to the size of the gridded beam with a taper, and the rightmost columns show the RMS when the beams are forced the other way from gridded to gridless. The super-uniform weighting is done with the SDE scaled cell implementation. The points to note from this table are

- The gridless RMS is almost always superior to the gridded RMS regardless of the direction of tapering
- The few exceptions to the trend come from the unbalanced array cases. It seems that the variations in individual weights are so large that the RMS is determined by how the tracks happen to line up with each other not by the random effects of a single track on a grid. It is not clear why this effect should favor the gridded weighting.
- The difference in RMS for the two weightings is greatest for the (super) uniformly weighted beams, and becomes less as the robustness is increased. At high robustness the difference is negligible.
- The difference is more significant for super-uniform weighting than uniform weighting.

Cov	FOV	Rob	Beam (/mas)	RMS GRD	RMS GL, Tap	Beam (/mas)	RMS GRD, Tap	RMS GL
VLA 3C48	1.00	Uni	0.1937	1.969	1.618	0.2007	1.860	1.533
VLA 3C48	1.00	0.0	0.2071	1.450	1.397	0.2102	1.423	1.370
VLA 3C48	1.00	0.5	0.2401	1.132	1.126	0.2418	1.127	1.121
VLA 3C48	0.25	Uni	0.1607	3.966	2.523	0.1703	3.273	2.156
VLA 3C48	0.25	0.0	0.1871	1.610	1.589	0.1911	1.540	1.520
VLA 3C48	0.25	0.5	0.2295	1.154	1.150	0.2333	1.138	1.134
VLA Snap 30	1.00	Uni	0.2162	1.530	1.065	0.2298	1.490	1.034
VLA Snap 30	1.00	0.0	0.2198	1.302	1.053	0.2309	1.279	1.031
VLA Snap 30	1.00	0.5	0.2321	1.073	1.023	0.2374	1.068	1.017
VLA Snap 30	0.25	Uni	0.2085	1.459	1.121	0.2172	1.419	1.106
VLA Snap 30	0.25	0.0	0.2113	1.224	1.111	0.2181	1.210	1.100
VLA Snap 30	0.25	0.5	0.2204	1.090	1.069	0.2245	1.085	1.064
VLBA	1.00	Uni	4.432	1.669	1.387	4.499	1.646	1.364
VLBA	1.00	0.0	4.673	1.379	1.296	4.694	1.375	1.291
VLBA	1.00	0.5	5.408	1.130	1.119	5.429	1.128	1.118
VLBA	0.25	Uni	3.737	2.581	1.822	3.917	2.342	1.680
VLBA	0.25	0.0	4.217	1.510	1.478	4.293	1.479	1.447
VLBA	0.25	0.5	5.168	1.157	1.152	5.221	1.151	1.145
+ VLA/GBT	1.00	Uni	4.815	4.979	4.348	4.808	4.993	4.359
+ VLA/GBT	1.00	0.0	6.563	1.663	2.070	5.994	1.796	2.173
+ VLA/GBT	1.00	0.5	8.013	1.195	1.356	7.207	1.283	1.399
+ VLA/GBT	0.25	Uni	4.085	7.428	5.732	4.151	7.172	5.553
+ VLA/GBT	0.25	0.0	5.934	1.745	1.900	5.526	1.893	2.044
+ VLA/GBT	0.25	0.5	7.692	1.201	1.276	7.094	1.275	1.330
+ Orbiter	1.00	Uni	1.188	18.629	19.119	1.185	18.633	19.130
+ Orbiter	1.00	-0.5	2.236	3.836	4.370	1.832	4.632	4.912
+ Orbiter	1.00	0.0	3.898	1.866	2.095	3.317	2.104	2.266
+ Orbiter	0.25	Uni	1.175	19.566	16.349	1.180	19.328	16.458
+ Orbiter	0.25	-0.5	1.886	4.516	4.701	1.757	4.759	4.931
+ Orbiter	0.25	0.0	3.389	2.159	2.264	3.180	2.264	2.348

Table 3.9: Gridded/Gridless RMS comparison

At this point, things have become complicated. We have several possible knobs which may be turned in order to control the beam shape. Both super-uniform weighting and robustness produce the typical “L” shaped RMS/resolution tradeoff curve, and to a lesser extent time averaging and thresholding do as well. As shown in the earlier plots, tapering down tends to produce a wider beam and usually better thermal RMS. Inverse tapering goes the other way, so we have yet another way of exploring RMS/resolution space. The question now is *“what is the effect of tapering from different starting points?”* Figure 3.34 addresses this for the specific case of the synthetic VLA Full Tracks data. The dashed lines are the same as the main traces in Figure 3.23 and are produced by varying the robustness. The point labelled 0 is no taper at all, and each small dot is a taper increment of $.01''$. That is, the first small dot above the 0 has been smoothed with a Gaussian of size $.01'' \times .01''$. The second with a Gaussian of size $.02'' \times .02''$ and so on. Every tenth point is circled.

Examining the traces parameterized by taper, we see that while the qualitative shape of the curve is much the same as that of robustness, the slope at a given point is often quite different. Tapering decreases the thermal noise when the combination of the starting point plus taper resembles natural weighting more closely than the starting point alone. Most arrays have sampling patterns strongly peaked at low frequencies, so a flat uniform weighting plus downweighting of the higher spatial frequencies yields a better match and lower thermal RMS than uniform weighting. The typical problem with tapering is that the Gaussian taper dies off too quickly. In order to match the shape of the central portion of the sampling pattern, a taper must be applied that is essentially zero at the edge of the pattern. By that point, the outer spacings are being thrown away completely. The high taper limit actually resembles that of the thresholding curve of Figure 3.24, and for much the same reason as there. This is responsible for the tapered trace being steeper than the robust trace in the uniform panel of Figure 3.34a. According to these plots, a taper is a bad way to work the RMS/resolution tradeoff when desiring a wide beam. The compensating factor which is not apparent from these plots is that the tapered low resolution beam is beautifully Gaussian, while less sensitive than an equivalent resolution robust beam.

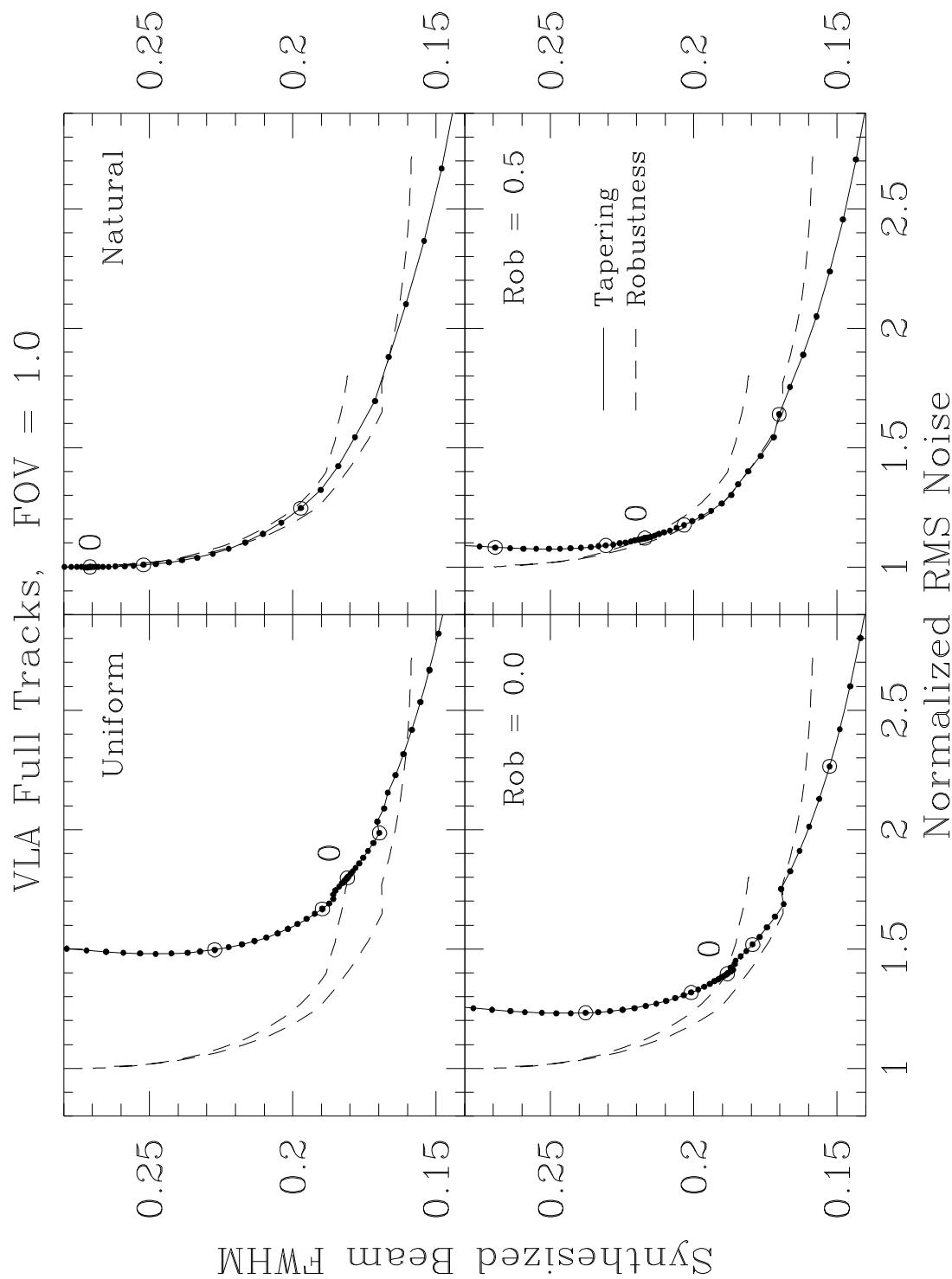


Figure 3.34a: How much robustness do we want, if we're going to taper? $FOV_w = 1$

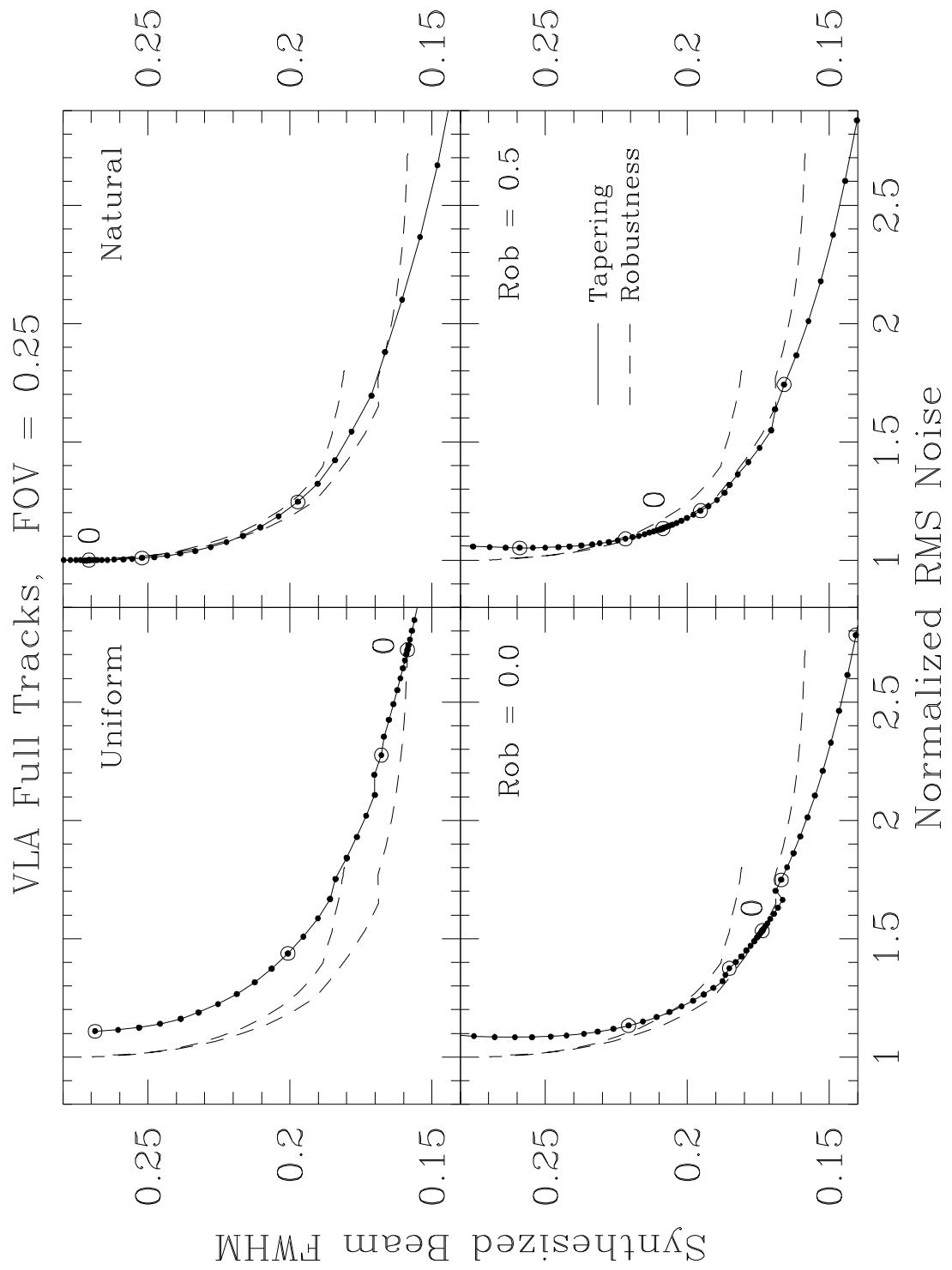


Figure 3.34b: Same as previous panel, but with $\text{FOV}_w = 0.25$

From the figures, it might appear that natural weighting plus an inverse taper is much equivalent to robustness. In terms of these two measures of beam quality is is, though again the measures plotted in these figures are not sufficient to differentiate between the character of the different beams. We look at some additional tabulated quantities in Tables 3.10 and 3.11, but at this level of detail there is no substitute for looking at more examples directly. The two tables attempt to answer the question “*How much robustness do we want, if we’re going to taper?*”, tabulating first the RMS and then the most negative sidelobes when a number of different starting points have been forced to a common beam size with a taper. The three beam sizes listed in the table as small, medium, and large are that of the gridded $FOV_w = 1/4$ super-uniform beam, the gridded uniform $FOV_w = 1$ beam and the geometric mean of the gridded uniform and naturally weighted beam. All forced beams are circular.

The most important conclusion of the first table is that “*some*” robustness is sufficient to gain the majority of the thermal sensitivity improvement. In most cases there is a minimum in RMS against robustness, and in others the naturally weighted starting point still provides the best thermal RMS, even after taper. But in all cases, the minimum in RMS is rather broad. A normalized robustness of 0 or 0.5 is reasonably close to the minimum value, and is substantially better than starting from uniform weighting.

The second table tabulates the most negative sidelobes instead of the RMS. It’s primary use here is to demonstrate why I haven’t often tabulated this quantity! There are a number of different competing effects contributing to this table, and the resulting values show few clear trends. More often than not, the naturally weighted beams have the smallest negative sidelobes, but that is usually the result of an extended shelf obscuring the worst of the near-in sidelobes. Occasionally it the uniform beam with the least sidelobes, or intermediate value of the robustness. The gridless beam usually but not always has a smaller negative sidelobe than the equivalent gridded case. For every trend in this table, there is also a counter example. The conclusion is that when using this many parameters to control the beam shape, there is no scalar measure substitute for direct examination of the beams and a measure of subjective judgement.

Cov	FOV	Rob	RMS (Gridded)			RMS (Gridless)		
			Small	Medium	Large	Small	Medium	Large
Small = 0.1607"								
VLA 3C48	1.00	Uni	3.075	2.059	1.585	2.616	1.688	1.315
VLA 3C48	1.00	0.0	2.579	1.668	1.304	2.557	1.620	1.248
VLA 3C48	1.00	0.5	2.643	1.599	1.158	2.661	1.605	1.153
VLA 3C48	0.00	Nat	3.031	1.813	1.184	3.031	1.813	1.184
VLA 3C48	0.25	Uni	4.192	2.308	1.316	2.588	1.647	1.190
VLA 3C48	0.25	0.0	2.411	1.563	1.171	2.405	1.546	1.167
VLA 3C48	0.25	0.5	2.543	1.555	1.139	2.580	1.566	1.137
Small = 0.2085" Medium = 0.2162" Large=0.2345"								
VLA Snap	1.00	Uni	1.583	1.544	1.486	1.100	1.070	1.031
VLA Snap	1.00	0.0	1.359	1.326	1.280	1.098	1.068	1.030
VLA Snap	1.00	0.5	1.140	1.112	1.076	1.090	1.061	1.024
VLA Snap	0.00	Nat	1.086	1.055	1.015	1.086	1.055	1.015
VLA Snap	0.25	Uni	1.443	1.415	1.370	1.126	1.112	1.103
VLA Snap	0.25	0.0	1.235	1.217	1.197	1.122	1.107	1.096
VLA Snap	0.25	0.5	1.119	1.101	1.083	1.100	1.081	1.060
Small = 3.737 mas Medium = 4.432 mas Large=5.679 mas								
VLBA	1.00	Uni	2.124	1.674	1.475	1.823	1.394	1.192
VLBA	1.00	0.0	1.887	1.459	1.263	1.817	1.380	1.168
VLBA	1.00	0.5	1.856	1.371	1.110	1.853	1.366	1.098
VLBA	0.00	Nat	2.017	1.454	1.092	2.017	1.454	1.092
VLBA	0.25	Uni	2.578	1.903	1.424	1.825	1.458	1.265
VLBA	0.25	0.0	1.821	1.435	1.231	1.782	1.404	1.212
VLBA	0.25	0.5	1.825	1.357	1.112	1.819	1.355	1.106
Small = 4.075 mas Medium = 4.817 mas Large=6.545 mas								
+ VLA/GBT	1.00	Uni	3.436	2.508	1.986	3.064	2.212	1.665
+ VLA/GBT	1.00	0.0	2.948	2.033	1.455	2.933	2.036	1.463
+ VLA/GBT	1.00	0.5	3.058	1.981	1.234	3.018	1.979	1.240
+ VLA/GBT	0.00	Nat	3.463	2.302	1.255	3.463	2.302	1.255
+ VLA/GBT	0.25	Uni	3.882	2.839	1.968	3.087	2.255	1.628
+ VLA/GBT	0.25	0.0	2.811	1.981	1.420	2.805	1.976	1.405
+ VLA/GBT	0.25	0.5	2.969	1.951	1.228	2.934	1.933	1.224

Table 3.10: Robustness/Taper tradeoff, RMS comparison

Cov	FOV	Rob	Negative Sidelobes (Gridded)			Negative Sidelobes (Gridless)		
			Small	Medium	Large	Small	Medium	Large
Small = 0.1607" Medium = 0.1937" Large=0.2444"								
VLA 3C48	1.00	Uni	-0.1951	-0.1149	-0.0501	-0.1844	-0.0945	-0.0383
VLA 3C48	1.00	0.0	-0.1798	-0.1017	-0.0456	-0.1816	-0.0886	-0.0341
VLA 3C48	1.00	0.5	-0.1791	-0.0869	-0.0285	-0.1786	-0.0848	-0.0309
VLA 3C48	0.00	Nat	-0.1400	-0.0632	-0.0423	-0.1400	-0.0632	-0.0423
VLA 3C48	0.25	Uni	-0.1597	-0.0528	-0.0342	-0.1723	-0.0760	-0.0341
VLA 3C48	0.25	0.0	-0.1906	-0.0979	-0.0325	-0.1874	-0.0990	-0.0337
VLA 3C48	0.25	0.5	-0.1825	-0.0902	-0.0326	-0.1854	-0.0944	-0.0330
Small = 0.2085" Medium = 0.2162" Large=0.2345"								
VLA Snap	1.00	Uni	-0.1370	-0.1309	-0.1196	-0.1119	-0.1035	-0.0865
VLA Snap	1.00	0.0	-0.1316	-0.1256	-0.1133	-0.1115	-0.1031	-0.0859
VLA Snap	1.00	0.5	-0.1197	-0.1091	-0.0913	-0.1082	-0.0995	-0.0818
VLA Snap	0.00	Nat	-0.0953	-0.0892	-0.0796	-0.0953	-0.0892	-0.0796
VLA Snap	0.25	Uni	-0.1518	-0.1466	-0.1370	-0.1644	-0.1596	-0.1489
VLA Snap	0.25	0.0	-0.1545	-0.1482	-0.1362	-0.1616	-0.1566	-0.1456
VLA Snap	0.25	0.5	-0.1440	-0.1367	-0.1216	-0.1435	-0.1376	-0.1241
Small = 3.737 mas Medium = 4.432 mas Large=5.679 mas								
VLBA	1.00	Uni	-0.1261	-0.0555	-0.0582	-0.1091	-0.0668	-0.0698
VLBA	1.00	0.0	-0.1174	-0.0561	-0.0589	-0.1080	-0.0654	-0.0669
VLBA	1.00	0.5	-0.0948	-0.0551	-0.0476	-0.0925	-0.0608	-0.0523
VLBA	0.00	Nat	-0.0782	-0.0606	-0.0459	-0.0782	-0.0606	-0.0459
VLBA	0.25	Uni	-0.1400	-0.0747	-0.0550	-0.1424	-0.0860	-0.0389
VLBA	0.25	0.0	-0.1313	-0.0674	-0.0406	-0.1342	-0.0775	-0.0350
VLBA	0.25	0.5	-0.1038	-0.0497	-0.0368	-0.1027	-0.0501	-0.0362
Small = 4.075 mas Medium = 4.817 mas Large=6.545 mas								
VLBA+	1.00	Uni	-0.1639	-0.0910	-0.0863	-0.1633	-0.0996	-0.1057
VLBA+	1.00	0.0	-0.1623	-0.0865	-0.0840	-0.1644	-0.0933	-0.1006
VLBA+	1.00	0.5	-0.1382	-0.0867	-0.0730	-0.1455	-0.0906	-0.0855
VLBA+	0.00	Nat	-0.1041	-0.0701	-0.0554	-0.1041	-0.0701	-0.0554
VLBA+	0.25	Uni	-0.1842	-0.1014	-0.0805	-0.1622	-0.0848	-0.0644
VLBA+	0.25	0.0	-0.1872	-0.0856	-0.0601	-0.1794	-0.0827	-0.0599
VLBA+	0.25	0.5	-0.1534	-0.0881	-0.0640	-0.1538	-0.0890	-0.0641

Table 3.11: Robustness/Taper tradeoff, Negative sidelobe comparison

In this spirit of direct examination, we show contours and slices of the VLA Full Track case as a function of robustness, where the fitted beam size has been held constant with a taper to that of the gridded uniform case. Compare this with the untapered plots of Figure 3.13. In general, there is less variation between the different beams, as the gross properties of the beam are largely held constant and only the second order effects allowed to vary. Again we see the minimum in RMS as a function of starting robustness. The usual wide shelf around the natural beam is largely suppressed, but still present at a level too large to recommend. Of the $FOV_w = 1.0$ series, the robustness of 1 or 0.5 isn't a bad choice, though one would only inverse taper like this if the highest possible resolution beam were required. The $FOV_w = 1/4$ series in the following plot simply shows that when one forces the beam to a sufficiently small target, there is little that can be done about ringing sidelobes.

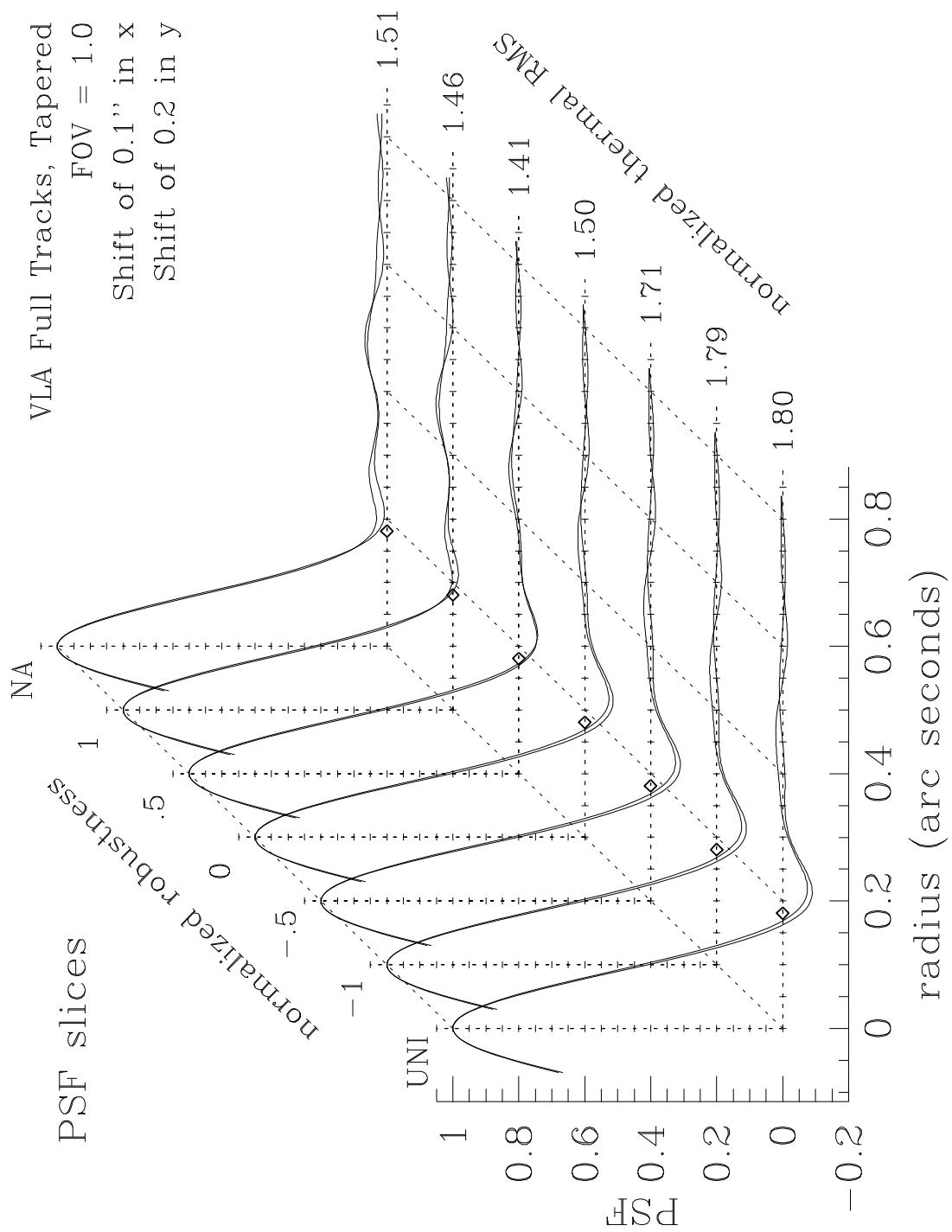


Figure 3.35a: PSF plots against robustness, Tapered, $FOV_w = 1.0$

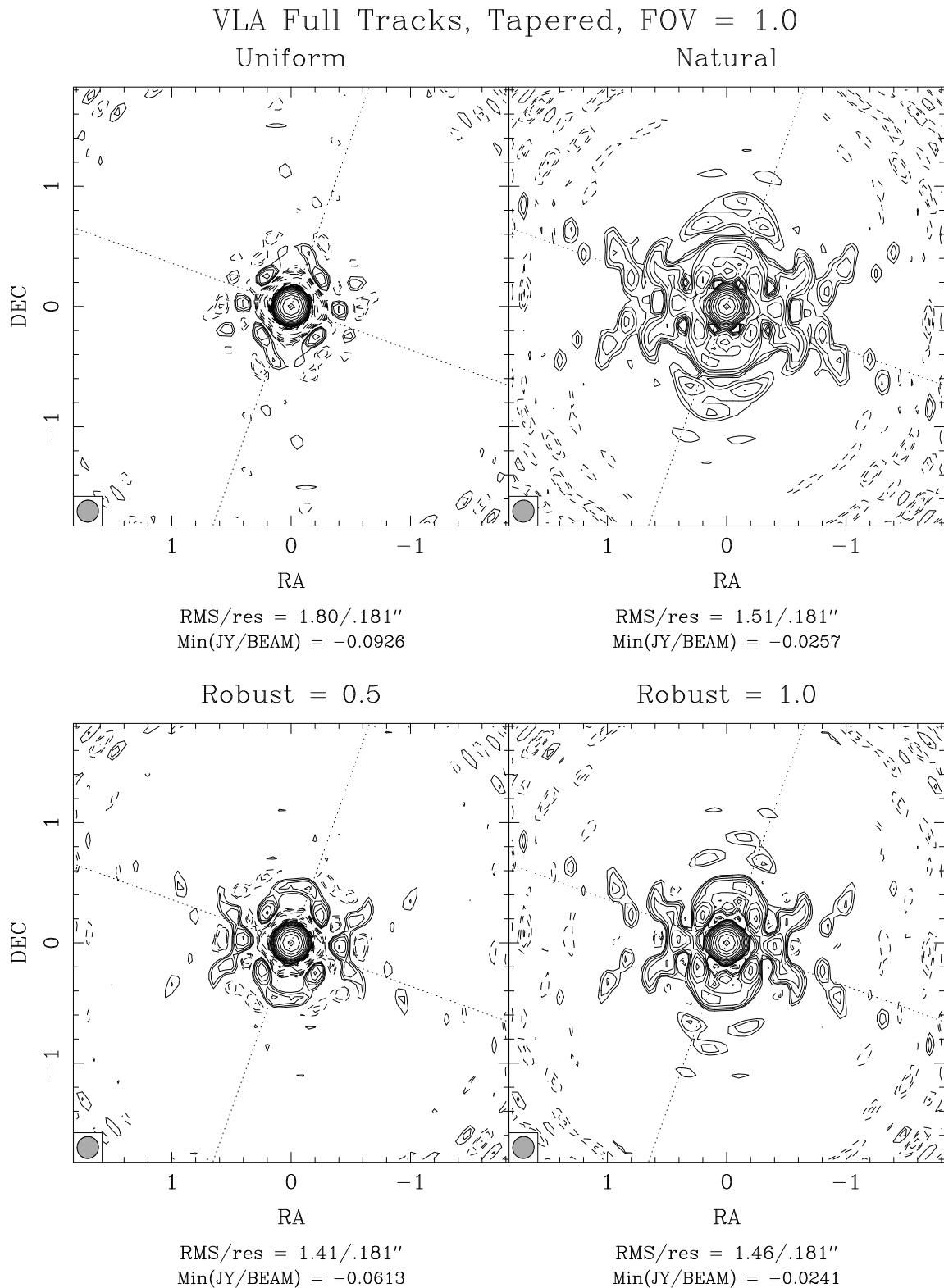


Figure 3.35b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

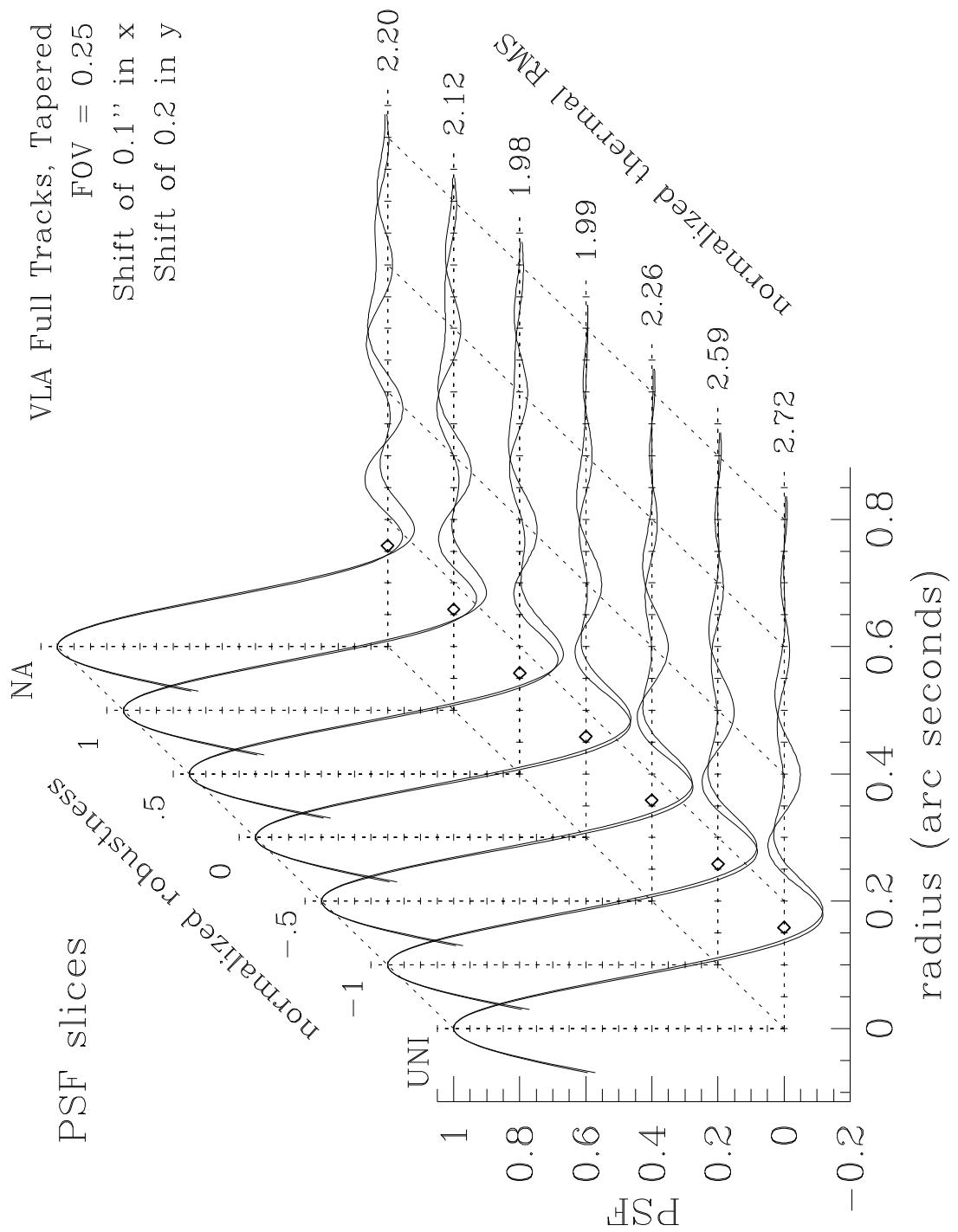


Figure 3.36a: PSF plots against robustness, Tapered, $FOV_w = 0.25$

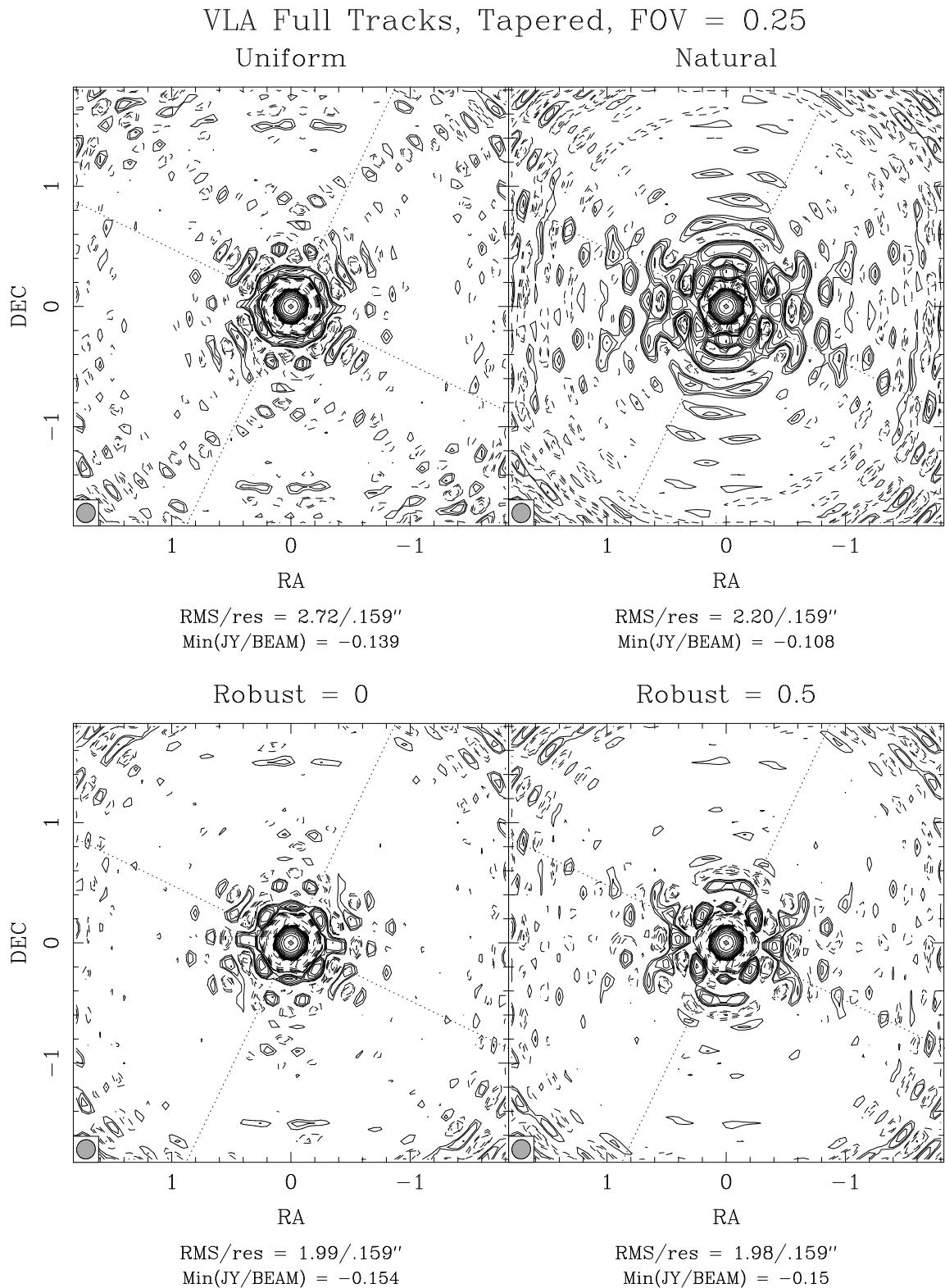


Figure 3.36b: Factor of $\sqrt{2}$ between contours, lowest is 1%.

We conclude this section with one last series of a common taper, combining a number of the different features discussed. The plots have been extended out considerably further than the rest in this chapter to show the far-off sidelobes which have been somewhat neglected. The first panel is the gridded, $FOV_w = 1$, robust=0.6 case from the same VLA Full Track data set. This is an excellent beam for general purposes, and unless there is a special requirement for high resolution, it is the one recommended for this data. The beam parameters are $1.086/.225''$, with a worst negative of $-.0203$. At this high a level of robustness, there is very little difference between the gridded and gridless weighting. The corresponding gridless case forced to this resolution is only $1.082/.225''$, with a $-.0216$ worst negative. It is not shown. The next panel is the naturally weighted beam forced to the same resolution with a taper. The only thing that this has bought us is a very slight increase in the thermal sensitivity; both the near-in sidelobe shelf and far sidelobes have worsened considerably. The previous beam is so good in thermal noise, that the extra sensitivity this has bought us is negligible. The gridded and tapered uniform beam is given next. The thermal RMS has risen considerably to 1.497, and the worst negative dropped slightly to $-.0322$, but of all the cases shown this has the best far-off sidelobes. The choice of this or the robust beam would depend on which features of the beam are limiting the final map. In a sensitivity limited situation, the robust beam is the better choice. In a particularly difficult deconvolution situation, this might be used. Alternatively, one could use the gridless uniform beam given as the last panel in the series. Without robustness, the gridless weighting does make a considerable difference in the thermal noise, which drops back to 1.222. The near-in negatives are slightly worse at $-.0364$. In general, the far-off sidelobes of the gridless uniform beam will be worse than the gridded uniform beam, as the later *is* the minimum sidelobe solution. It's usually a judgement call whether this is a better choice than a robust beam. In this case, I would probably prefer one of the two previous good choices, but I would try all of them if it was a case where the deconvolution problem was difficult.

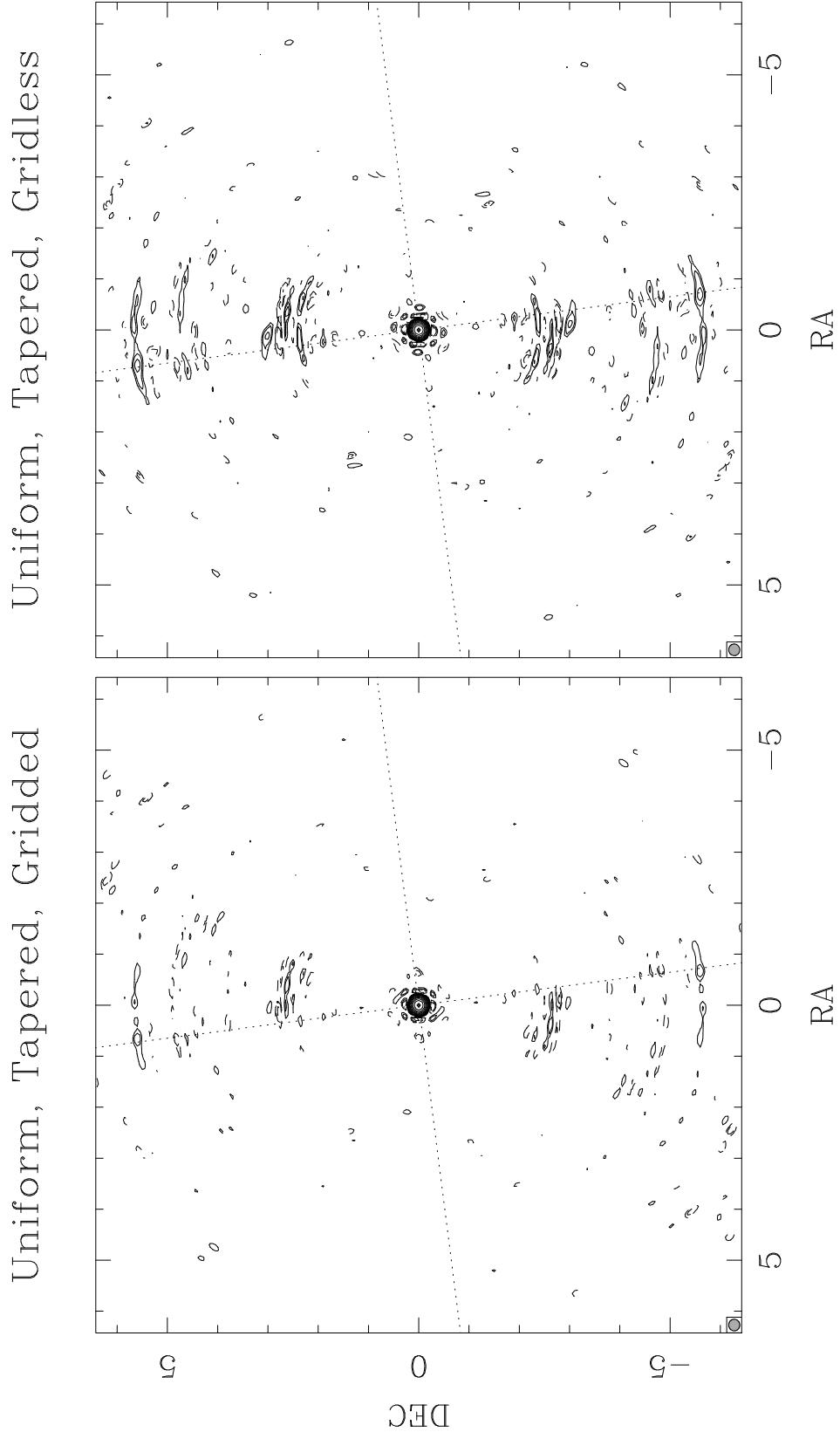


Figure 3.37a: Example PSFs tapered to a common resolution. Factor of $\sqrt{2}$ between contours, lowest is 1%.

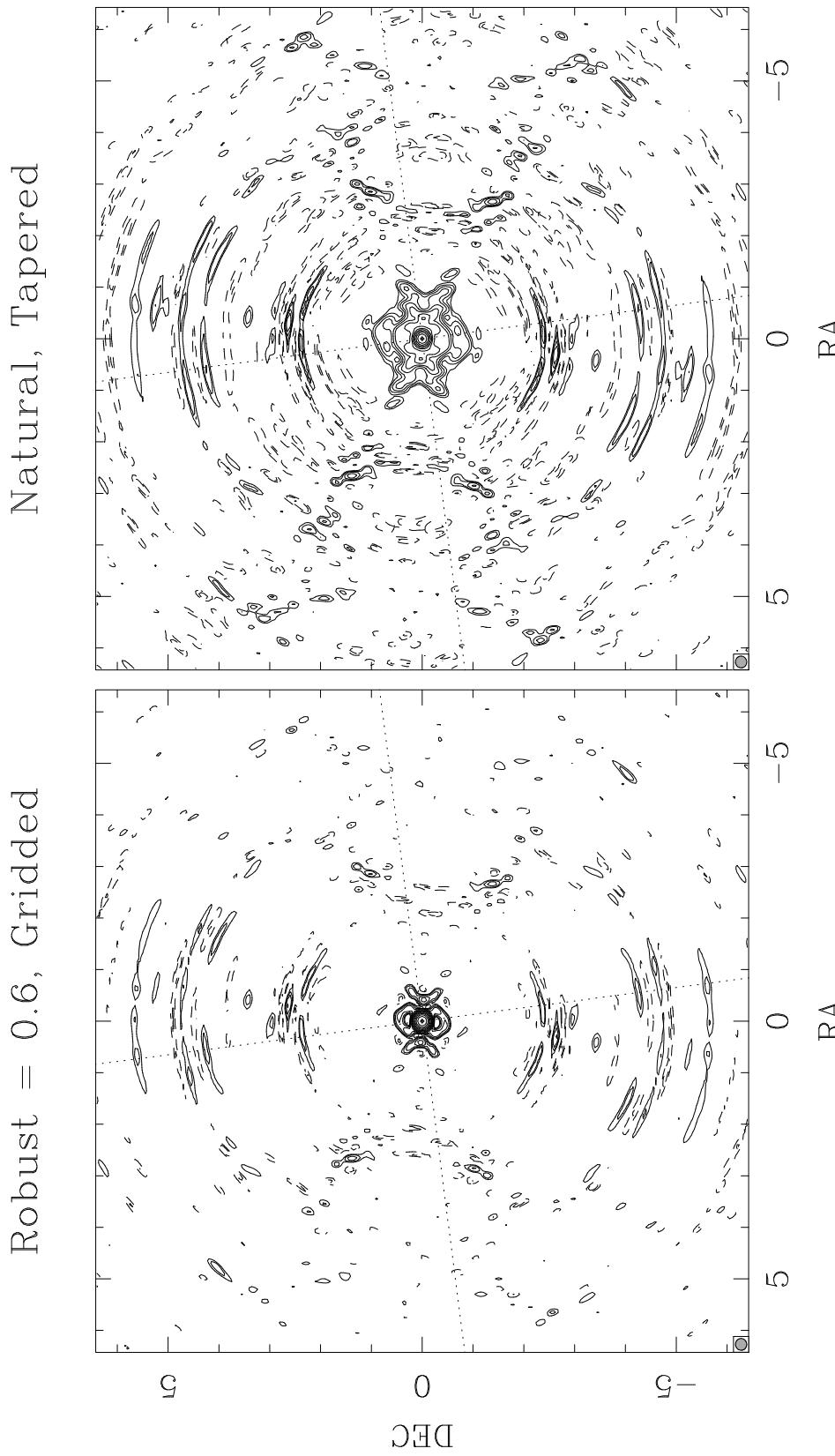


Figure 3.37b: PSFs tapered to a common resolution, continued. Factor of $\sqrt{2}$ between contours, lowest is 1%

3.11.1 VLB (Square Root) Weighting

The badly unbalanced arrays sometimes found in VLBI can lead to nearly unusable naturally weighted beams. This has lead to a tradition in the community of arbitrarily downweighting the more sensitive baselines before the density weighting. Common choices are taking the square root or fourth root of the snr weights, or even setting them to unity. The Caltech VLBI package and SDE are the most sophisticated in this regard, and allows one to raise the density weights to an arbitrary power before use. While certainly better than nothing, this is a rather drastic approach to curing the problem. Modification of the density weights in this manner is really only appropriate for controlling the relative weights between baselines that combine together into the same gridded visibility. Modification of the weights for imaging purposes is best achieved by the operating on the gridded visibilities with a density weight or taper. Forcing one type of the weight control to do the job of the other results in a poor use of the available signal and a poorer thermal sensitivity than necessary.

The robustness tradeoff curves for the two unbalanced array cases are given in Figure 3.38. In each, there are traces from the unmodified snr weights, the weights modified by taking their square root, their fourth root, and by setting all snr weights to unity. That is, the four traces correspond to snr weight exponents of 1.0, 0.5, 0.25, and 0.0. In addition, there are traces for the both the uniform and $FOV_w = 1/4$ super-uniform weighting cases. The imaging parameters are again those from Table 3.7.

Modifications of this type to the weights makes relatively little difference to the uniformly weighted beam. Clearly the beam shape will change essentially not at all, as all occupied cells are reweighted to unity by the density weighting and reweighting cannot change which are occupied. It is interesting that the RMS changes so little, however. There is more action in the natural weighting limit. As the snr weights are progressively flattened, the natural limit traces out the curve indicated by the four open diamonds in Figure 3.38a. Of interest is that this limiting curve is very *un*-favorable in terms of the RMS-resolution tradeoff. As the qualitative shape of the tradeoff curve does not change greatly for this case, moving the limiting point into an unfavorable regions of the RMS-resolution plane pulls the entire curve into unfavorable regions compared to the corresponding cases of the unmodified weights. Similar behavior is seen in the orbiter case, though it is quantitatively quite different.

As demonstrated before, the RMS & beamwidth do not tell the whole story about beam properties. The main reason for the VLB weighting in the first place is the desire to avoid the sidelobes caused by a few baselines dominating the sensitivity. Again we examine some representative beams in detail. Figures 3.39 and 3.40 are a repeat of Figures 3.21 and 3.22, with the addition of square root modification of the snr weights. Clearly the sidelobes in Figure 3.39 are much improved over those in Figure 3.39, towards the natural weighting limit. The modified naturally weighted beam has turned the one dominant sidelobe into an extended shelf at somewhat lower magnitude. While certainly not an

attractive beam, it is clear why the practice of weight modification was started. Comparing beams at fixed values of the robustness, it might appear that the square root modification allows us to work closer to the naturally weighted limit. The weight modification has actually changed the robustness scale somewhat. The unmodified beams at a normalized robustness of R are roughly comparable to the modified beams at a robustness of $R + .5$. When we compare the appropriate beams, we find that at the common values of the resolution the unmodified beams do substantially better. For instance, at $R = 0.5$, the unmodified beam has an RMS/resolution of 1.20/8.02 mas. At $R = 1.0$, the modified beam is 1.59/8.17 mas. The sidelobes of the unmodified beam are at least as good as the modified one, so in all respects it is a better choice for a high sensitivity beam. The situation is similar for the orbiter case as exemplified by an unmodified beam of 3.84/2.27 mas at $R = -.5$ compared to the modified 4.54/2.32 mas at $R = 0$.

Robustness is clearly an improvement over the VLB style weighting when imaging well-calibrated data, and even an improvement over VLB weighting plus robustness. There is still good practical use for the weighting during calibration, however. The snr weights really only reflect the reliability of the data when the calibration is under control, and certainly do not at the start of the calibration loop. If the visibility phases are completely random, then the snr weights have *nothing* to do with the reliability and an appropriate choice for the first iteration of self-calibration might be all snr weights equal. The optimum weights and strategy for a real calibration can be a very complicated question, and it has not been investigated here in detail. As a general principle, flattening the weights towards the beginning of the calibration process and returning to the unmodified snr weights plus robustness seems like a good strategy.

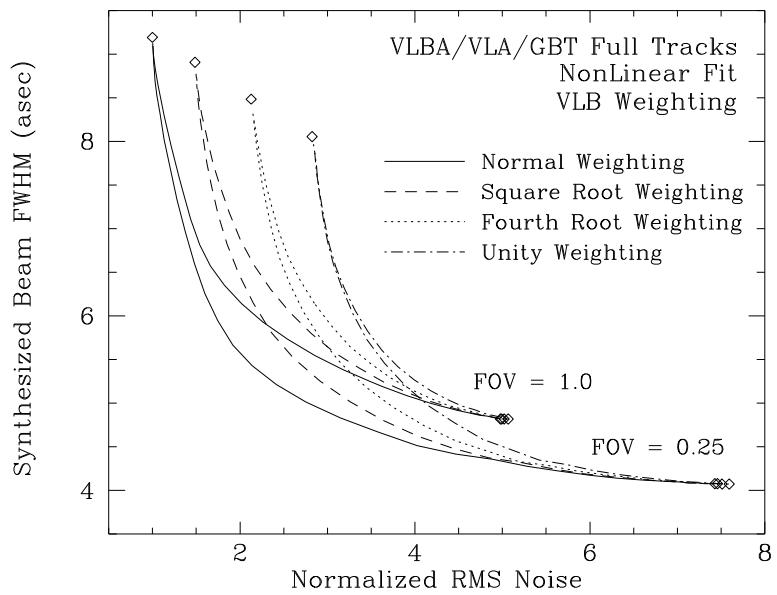


Figure 3.38a: Robustness tradeoff curves, VLBI weighting comparison.

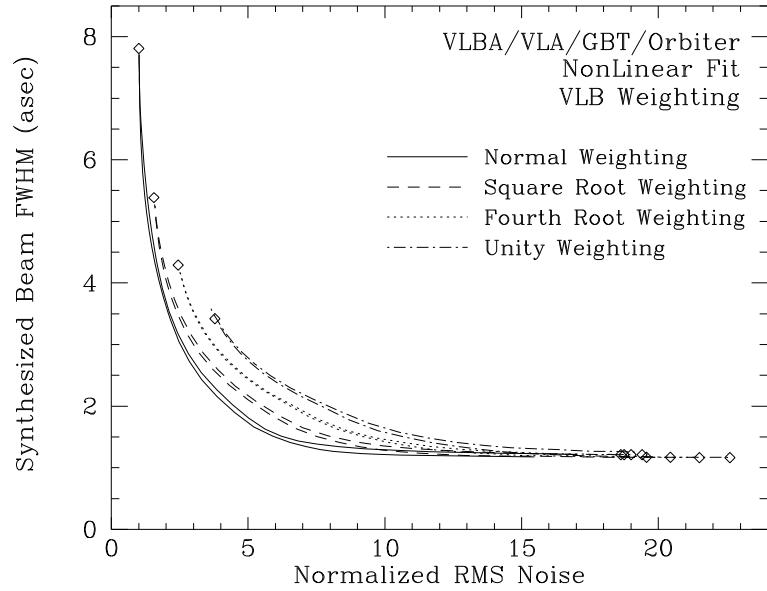


Figure 3.38b: Same as previous, but for the extremely unbalanced Orbiter case.

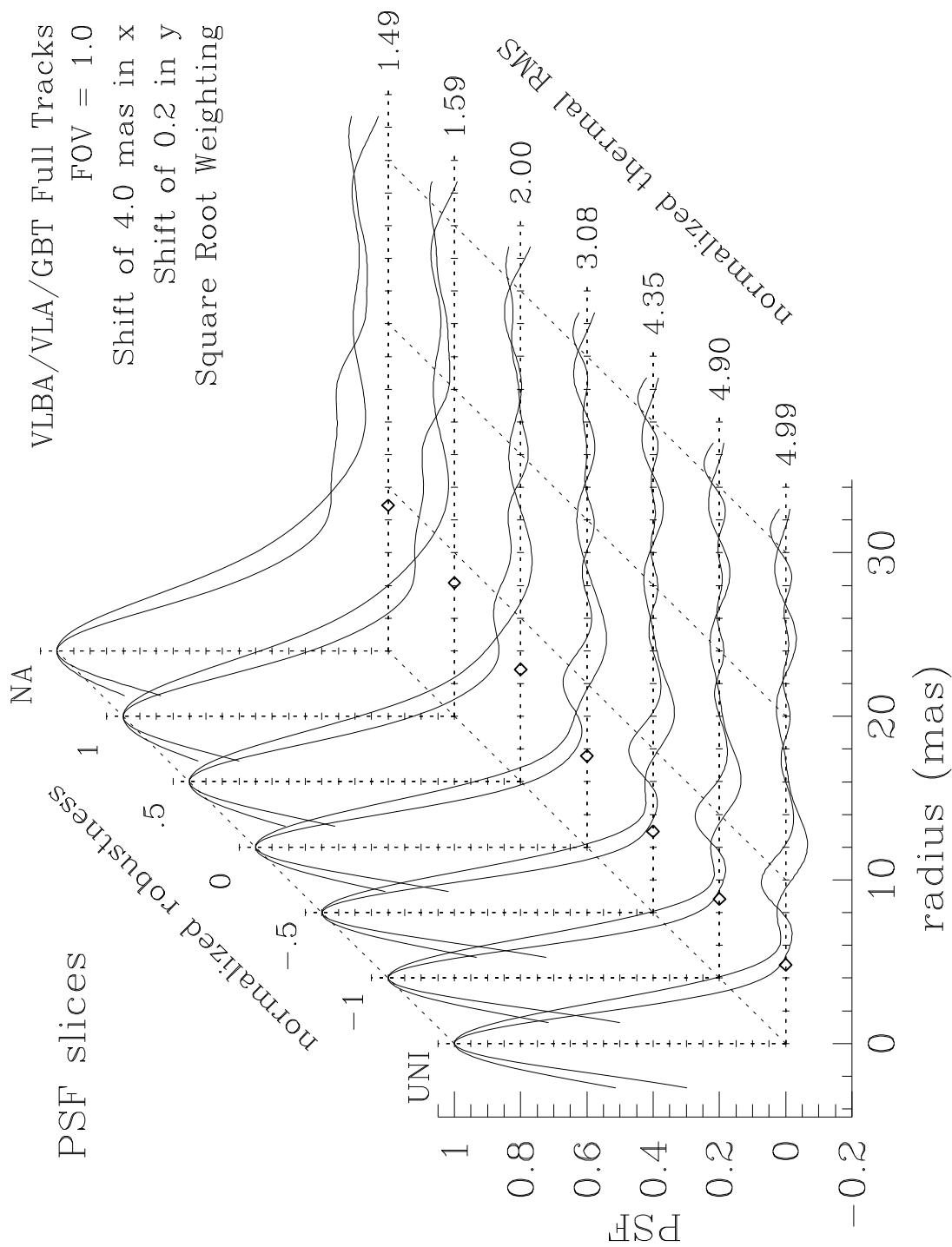


Figure 3.39a: PSF plots against robustness, VLBA/VLA/GBT Full Tracks (VLB weighting).

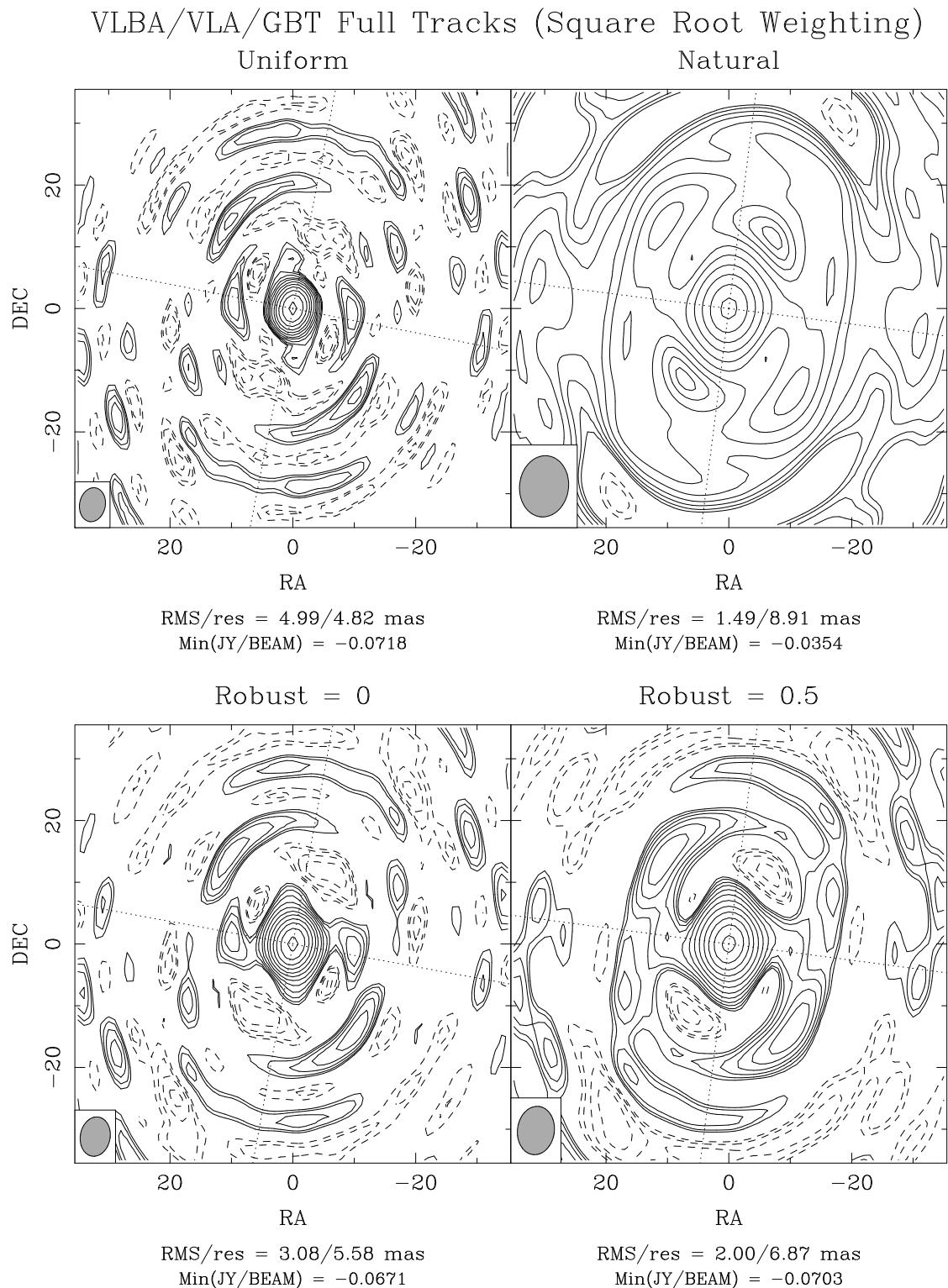


Figure 3.39b: Factor of $\sqrt{2}$ between contours, lowest is 2%

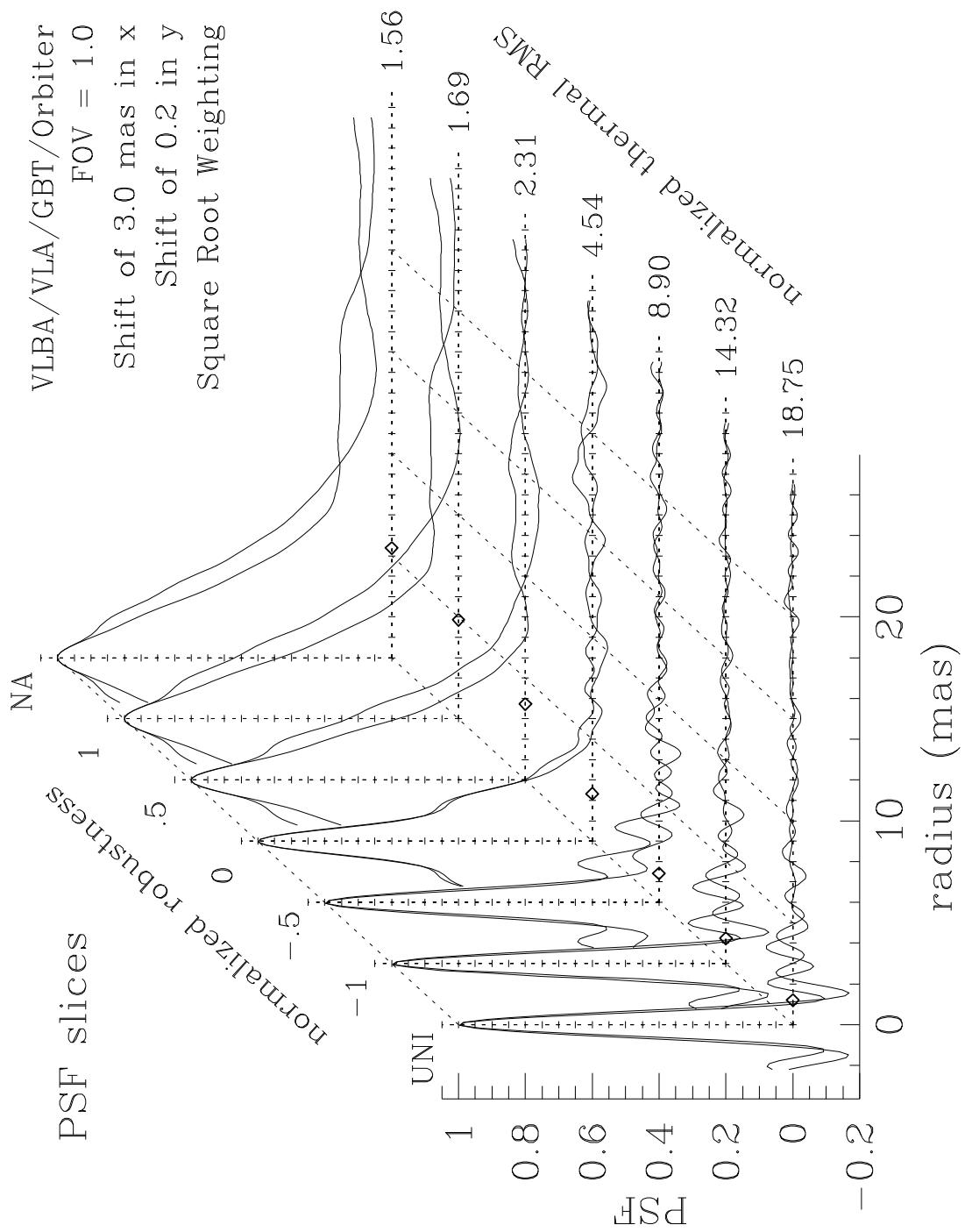


Figure 3.40a: PSF plots against robustness, VLBA/VLA/GBT/Orbiter (VLB weighting).

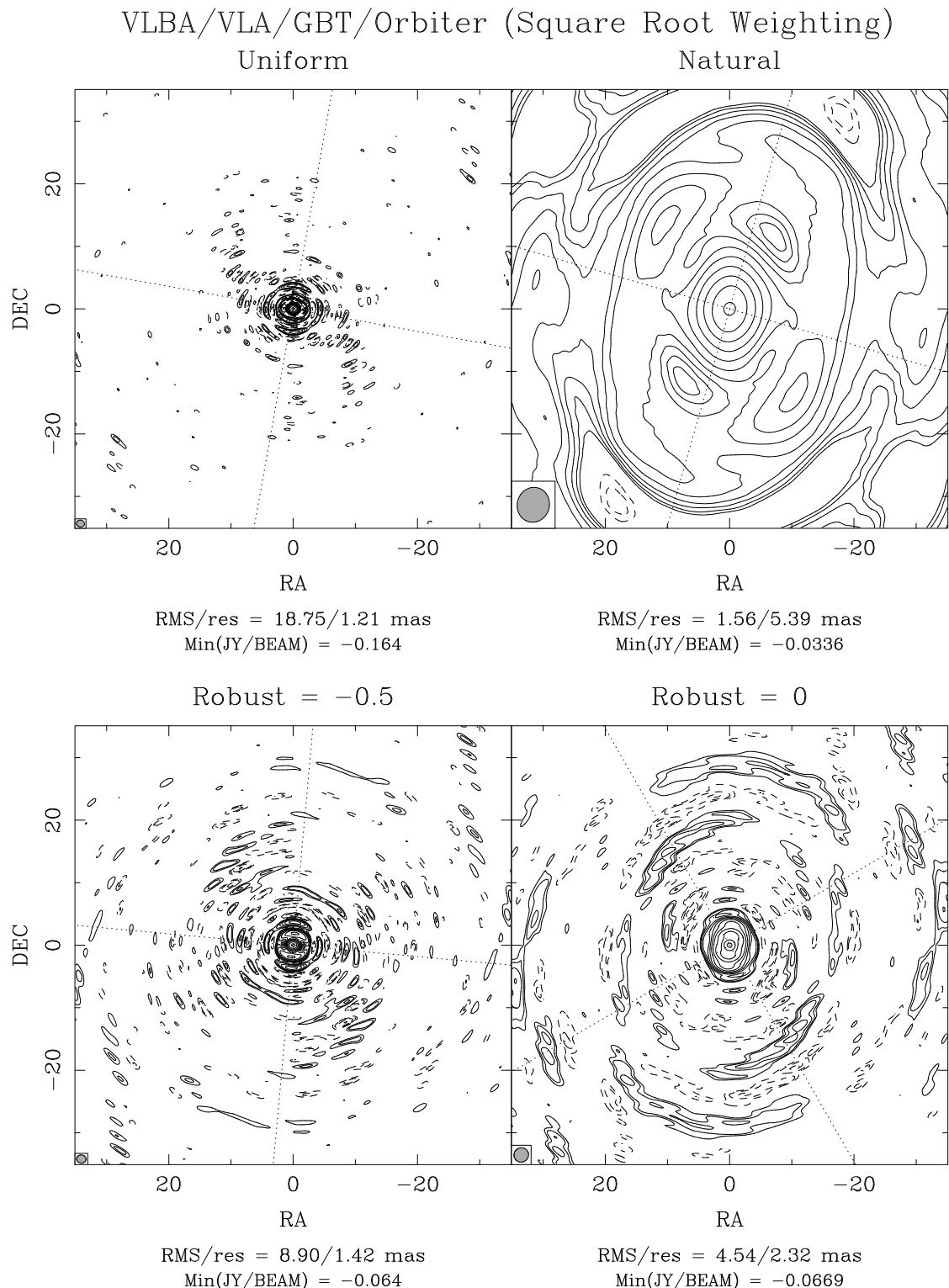


Figure 3.40b: Factor of $\sqrt{2}$ between contours, lowest is 2%

3.12 Future Work

3.12.1 Azimuthally Varying Robustness

The sidelobe difficulties in the AT case and to a less extent in the 3C48 case suggest that an ad hoc extension of the robust weighting formalism would be useful to create a beam with the best possible fit to a Gaussian. An obvious approach is to vary the effective degree of robustness as some function of azimuthal angle, in an attempt bring all sidelobes though zero simultaneously as the robustness is varied. This approach loses the aesthetic appeal of the minimization formalism, and it creates difficulties for programs which find the robustness automatically, since the search space is expanded to several dimensions. But for certain sampling patterns it might help considerably.

This was very briefly explored in SDE. An elliptical extension of robustness was considered, with the effective robustness at each pixel in the gridded visibility plane being proportional to the ‘radius’ of an ellipse at the position angle of the pixel. Unfortunately, the definition of normalized robustness goes both positive and negative. Even if the radius were used as a logarithmic parameter, the relative weighting of points at different position angles would depend strongly on the details of the normalization. Consequently, it was simply decided to have the effective robustness vary linearly with position angle between a minimum and maximum along orthogonal robustness axes. That is, the effective robustness is a saw tooth function in position angle, with a period of 180° . This choice does have the property that the mean robustness averaged over a circular region in the u - v plane is simply the arithmetic mean of the robustness along the two primary axes.

Figure 3.41 demonstrates that this experiment was not entirely successful. Essentially, the first sidelobes of the beam have been roughly forced to zero at four points in azimuth instead of two, and even that not terribly well. The functional form selected for robustness as a function of azimuth is a disappointingly poor fit to that necessary for a nice beam at all values of azimuth. A possible solution to this problem would be to abandon an analytic form in azimuth all together, and simply specify the azimuthal robustness as a sequence of values, possibly interpolated by a spline. The problem then becomes how to specify such a list. One could hook such an arbitrary 1-D robustness into a numerical minimizer against the fitted Gaussian in the style of Section 3.9, and use finite differences to approximate the necessary derivatives. If such an approach were used, one would presumably greatly oversample the beam to increase the number of pixels in each azimuthal bin, and hold the weighting fixed with FOV_w . The problem of guiding such a minimizer to an approximate global minimum is nontrivial, but it doesn’t seem entirely intractable. More work could be done here.

3.12.2 Thermal Noise Beam and Visualization

The ratio between the gridded snr weights and the total gridded weights determines the degree by which thermal noise at a given spatial frequency is amplified. In analogy

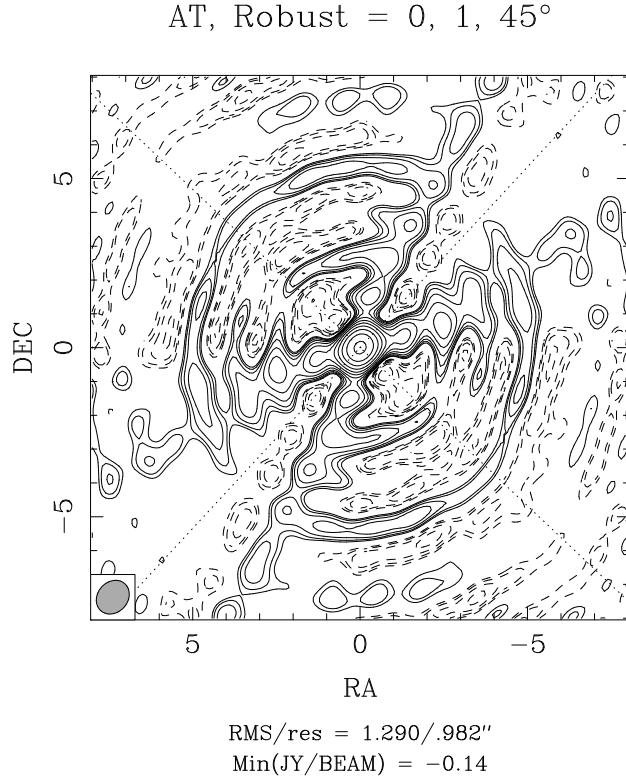


Figure 3.41: Azimuthally varying robustness. The effective robustness is 0 along a position angle of 45° and 1 along the orthogonal direction, and varies linearly with position angle in between.

to the calibration error beam occasionally used to discuss the morphology of antenna-based calibration errors, one might hope to define a thermal noise beam which in some sense will give predictions about how the noise is distributed. Unfortunately, we don't have the phase of the thermal noise at each gridded weight, all we have is the magnitude. The practical effect of this is that we can't say that the effect of the thermal noise amplification is a convolution with any particular beam. We can, however, just simulate the thermal noise. The eye is pretty good at picking out structure, and if there is structure in the simulated noise that resembles structure in the map, you've almost certainly amplified the noise at some frequencies beyond reasonable limits. Examining this ratio in the Fourier plane is of course the most precise way to tell. A good mapping program should print the maximum and minimum visibility weights, both before and after density reweighting. It would seem reasonable to also print the maximum and minimum value of this amplification as well. My intuition is that we are not making as good use as we might of the spatial information about the noise amplification. There is probably room for progress here.

3.12.3 Generalized Windows and Minimizing Source Sidelobes

It was shown in Section 3.3.3 that uniform weighting can be derived by minimizing the RMS sidelobes over the dirty beam. Later, it was shown that super-uniform weighting results from scaling the region over which the sidelobes are minimized. The region may be larger or smaller, but it is still rectangular. Bob Sault (1984) describes how the minimization criterion for the beam sidelobes may be generalized to include an arbitrary window function. That is, his method minimizes

$$\varepsilon^2 \equiv \sum_{i,j} q_{ij}(b_{ij} - m_{ij})^2$$

where q is a completely general windowing function, b is the dirty beam, and m is the ideal dirty beam.²¹ The window specifies not only where the sidelobes are to be minimized, but also how important the minimization is as a function of position. The resulting equations can be solved in an iterative fashion, with the more accurate of the methods he describes typically requiring 40–60 FFTs, and a slightly less accurate version which typically requires but 2 FFTs. Several examples of windows were presented: two central disks, a small patch centered on the worst off-peak sidelobe, and the autocorrelation of the small patch. (All window are symmetric around the origin.) The off-peak patch was the most impressive, and it successfully reduced the ε^2 criterion by more than a factor of seven over the uniformly weighted beam.

A detailed cross-comparison of these techniques with robustness has not been done. In particular, little is known about the thermal noise characteristics of the Sault optimally-weighted beams, beyond the obvious relationship to super-uniform weighting when a central window is used. Since there is no criterion in the Sault optimization which includes notice of the thermal noise, I suspect that it will suffer the usual problems with gridding irregularities and the consequent loss of sensitivity. The necessary equations to incorporate robustness into Sault’s formalism has been derived. The implementation appears to be straightforward, but there are no immediate plans to do so.

The Sault formalism is impressive in its sheer flexibility. When used with round constant windows centered on the origin it seems to be roughly comparable to other beam shaping techniques. But there are no other techniques extant for minimizing a single given sidelobe. Sault also presents an optimization strategy for minimizing the sidelobe response of the source at a given location in the dirty map rather than the dirty beam, but comments that the resulting equations are difficult to solve. In spite of the difficulty of the problem,

²¹The written description is slightly ambiguous. Notice that the obvious way of doing the differentiation in the image plane is wrong, as there are more degrees of freedom in the dirty beam than there are in the weighted visibilities. The minimization is consequently a constrained rather than an unconstrained problem. As it happens, doing the calculation properly in the Fourier plane or improperly in the image plane both return exactly the same result.

this appears to me to be the most promising avenue for further research in the area. In this dissertation, I have unified two distinct properties of the beam, resolution & sensitivity. In principle, Sault's work addresses sidelobes of the source. As computers and algorithms become more advanced, it might be hoped that future techniques could address all of these issues in a single optimization.

Finally, note that one might even use a different weighting in different regions of the dirty map. The algebraic deconvolution techniques discussed in Chapter 4 do not require a shift-invariant dirty beam, though they are limited at present to unfortunately small images. In the medium to long term future, it is possible that we will deliberately abandon the Fourier convolution equation with its shift invariant dirty beam in order to take full advantage of optimal weighting. Particularly if an efficient means can be found of solving the optimal weighting equations for a single pixel of the dirty map, a different weighting could be used for each pixel. The resulting transformation from visibilities to image plane is no longer a Fourier transform, but it is still a linear transformation and invertible by the most general deconvolvers. The NNLS algorithm from Chapter 4 is sufficiently successful that it is not clear the need exists for such heroic measures in weighting. But if ever the need should arise for ultra-high dynamic range or ultra-high-fidelity mapping of compact objects, this is at least a plausible avenue for future research.

3.13 Summary

I think that I've taken optimum weighting by adjusting parameters of the dirty beam into a region of diminishing returns. There may be better schemes, but the techniques described here work well enough that there is not tremendous room for improvement. As has been stated before, there are a lot of ways to get to roughly the same place when tuning beam parameters. The addition of robustness and a generalized taper adds sufficient flexibility that I can't help but think that they will be able to roughly approximate the results of most other simple reweightings on the gridded snr weights. Significant future results will likely come from generalization of the weighting to include some optimization criterion not considered here.

I know that the interaction of all these parameters and the characteristics of the sampling is confusing, and in some sense I've only made things worse by introducing yet another knob to play with. But I hope I've added a useful one, and also helped to clarify some of the interrelationships between existing parameters.

3.13.1 Imaging Recommendations

- For long-synthesis sensitivity-limited observations, the intermediate-resolution minimum-sidelobe beam is a very good general purpose choice.
- Stick with the shortest integration time that you can stand, and let robustness and gridless weighting do the optimum combination of visibilities for you.
- Super-uniform weighting can be very useful and should be explored in any project that uses uniform weighting. It's particularly important for snapshots and detailed beam shape control. It is also often useful to use a higher degree of super-uniform weighting and bring the beam back to a reasonable RMS with robustness.
- If you don't want to go that route, though, time averaging does have several useful features. It is similar in effect to a small amount of robustness in many cases, and it does provide some immunity to catastrophic gridding errors.
- Moderate time integration and super-uniform weighting can produce nearly as good a beam for VLA Snapshot coverage as robustness.
- Robust weighting is certainly not the be all and end all of weighting. There are many more things that could be done. This just happens to be easy, useful, and has an aesthetically pleasing derivation which leads to a more qualitative understanding of the properties.

3.13.2 Software Recommendations

The tradeoffs discussed are complicated, and it is clear that no single set of imaging strategies or parameters will suffice for all purposes. The tools should be made available for the astronomer to work properly. Roughly in order of priority, I recommend:

- Mapping programs should always calculate and display the normalized sensitivity loss due to weighting
- Implement gridded robust weighting, preferably including the normalized form. It's easy, and potentially very useful.
- Visualization tools are critical to allow the user to explore the imaging parameter space, robustness in particular but also FOV_w . This is the area most lacking in current software. Even the figures in this dissertation were produced with hand edited shellscripts. A single program should allow the user to explore the considerable parameter space of weightings, and plot the result in at least as much detail as here.
- Mapping programs should allow examination of the ungridded weights. All programs should allow printing of weighting statistics, (min, max, average, dispersion) of w_k and $D_k w_k$ and possibly $T_k D_k w_k$ as well.
- Encourage the use of real units in noise calculations, via ΔS_{uw} . Visibility averaging tasks can and should calculate this number from internal scatter in the data. We can do much better in routine estimation of actual thermal noise than is done currently.
- Mapping programs should include a full elliptical (inverse) Gaussian taper. Suggestions for implementing this are given in Appendix B. Print the statistics of the taper.
- Gridless weighting is handy to have for a number of reasons. It is useful for the highest sensitivity maps, in combination with any other weighting schemes. It is particularly useful for snapshot observations. It allows for easy implementation of sub-uniform weighting and is a good debugging aid for diagnosing gridding problems. It also has the convenience of being a single switch that the astronomer can throw to get ‘some’ sensitivity improvement from uniform weighting, even if robustness is not used.
- Examination of the gridded weights is useful. Due to the presence of the convolution function, this is not the same as back-transforming the dirty beam. Simple statistics of the gridded weights are sufficient, but a gridded weight image written out by the mapping program is better.
- Beam forcing is more for convenience than anything else, in that humans can generally guide a program to a reasonable answer fairly quickly. But if the generalized taper is available at all, this is so easy to implement that there's no excuse not to. Spectral line observers in particular will appreciate this.

- Automatic robustness selection by maximizing the best fit Gaussian is also nice, but the astronomer will likely be able to do as well by eye as by an automated procedure. Again the primary advantage is convenience.
- Post-hoc simulations of an existing observation that takes proper account of the snr weights should be easy and convenient. (Good post-hoc tools are useful in the context of weighting, but absolutely critical later on for diagnosing deconvolution instabilities.)

Chapter 4

Algebraic Deconvolution

“This chapter is devoted to the use of linear algebra applied to the image restoration problem.” (Andrews and Hunt, 1977, p. 147)

This quotation begins Chapter 8 of the rather prophetic book, *Digital Image Restoration*. The quotation served the original authors well; this was the book which largely introduced the modern concept of analyzing image restoration problems in terms of linear algebra. As the quote applies as well to my chapter as to theirs, I give their opening sentence one more use, some 18 years later. It is meant to underscore the fact that there is little fundamentally new in what I am about to describe. Indeed, singular value decomposition is a device from classical linear algebra, even if the first numerically stable algorithm to solve for the decomposition on a digital computer dates back only to 1965, (Golub and Kahan, 1965). The NNLS algorithm for the constrained least squares problem which we will make heavy use of later is from a book of a similar era, (Lawson and Hanson, 1974).

If the techniques are not new, then what is?

There is one fundamental distinction between the methods of Andrews and Hunt and those used here: In their methods, linear algebra was primarily used to analyse the problems — but most often the problems were reduced by clever manipulation of the matrix representations to more tractable ones in the frequency domain. Rarely were the problems solved directly, as the systems of equations were too large. While occasionally mention was made of the particular problems found in astronomy, it was clear that the authors were thinking in terms of the more usual picture processing. With every pixel filled with flux in varying degree, the general restoration such images is very difficult and one must resort to approximation, statistical treatments and cleverness. By contrast, here we will blunder through the problems using relatively simple and direct methods. Like the rest of this dissertation, the thrust of this chapter is largely pragmatic. There will be very little detailed analysis of the methods, some general discussion of why they work, and several examples demonstrating that NNLS in particular works *very* well for some problems. The previous chapter on weighting was a fairly thorough treatment. A few simple ideas were taken essentially all the way to their logical conclusions. Most of the obvious consequences were examined at least briefly. In comparison to that work, this chapter is a very promising beginning, but most emphatically far from complete.

Interestingly enough, there seems to be a convergence of research in the astronomical community towards algebraic methods of deconvolution, coming from quite different directions. I believe this chapter to be the first application of NNLS deconvolution to radio-interferometric imaging. In this field, the images were already too large for the computers of the day when NNLS was popularized as a model-fitting technique. It has only been recently realized that the overlap between computationally feasible images and scientifically interesting sources requiring high precision was large enough for algebraic techniques to be useful. The radio community has had to look backwards somewhat. The images that can be processed in this way are fairly small by the standards of the field, and the formalism is in the somewhat unfamiliar terms of model-fitting.

The high-energy astrophysics community is developing from the other direction. In the not too distant past, their detectors were of such low angular resolution that model-fitting was the only sensible type of processing to be done, and it was the image processing problems that were unfamiliar. To someone raised in a culture used to sophisticated modelling of complicated instrumental response and biases, it would seem only natural to continue to use such tools as the experimental resolution increased and the detectors became true imaging systems. And now, unlike 18 years ago, modern computers are largely up to the task of direct solution of the convolution equation. It will be interesting to see how similar the solutions become, starting from such different beginnings.

4.1 Representation

Representation issues were discussed in Section 2.3, where the continuous equations $I^D = I * B$ were forced into agreement with a dirty model representation $C * B$ at a collection of discrete points in the image plane, $\ell_{I,i}$, for $i = 1 \dots N_\ell$. A component-based representation of C was arbitrarily selected, using scaled and shifted replications of an arbitrary pixel function P . The locations of the model components are $\ell_{C,j}$, for $j = 1 \dots N_C$. (The usual form of the model chooses $P(\ell) \equiv \delta(\ell)$.) The convolution of C with B yielding I^D was discretized into a linear system of the form $\mathbf{Ax} = \mathbf{b}$. In matrix form, the elements of the matrix \mathbf{A} are samples of the dirty beam B , the elements of the vector \mathbf{b} are samples of the dirty image I^D , and the elements of the vector \mathbf{x} are the magnitudes of the delta-function model components of C . If the generalized pixel representation is used, the \mathbf{A} matrix is formed instead from samples of the modified dirty beam $B' = P * B$ and the unknown vector \mathbf{x} contains the coefficients of the pixel function rather than the magnitudes of the delta functions.

In nearly all interferometric applications, the matrix \mathbf{A} will be singular. There will either be none or infinitely many solutions to the system $\mathbf{Ax} = \mathbf{b}$, depending on whether or not \mathbf{b} is in the range of \mathbf{A} . (In normal use, \mathbf{b} will in the range of \mathbf{A} by construction, though roundoff or approximation error in the calculation of the dirty image and beam may violate this at a very low level.) The problem of deconvolution, then, becomes how to select an appropriate solution from the infinitely many that satisfy or nearly satisfy the only equations that we have.

The only possible way to do this is to introduce additional information or beliefs that we have about what a “correct” answer looks like. As argued before, the only connection between I and C is that they solve the same convolution equation and yield the dirty map when convolved with the dirty beam. There is no particular guarantee that I will even be representable in the particular form we have chosen for C , and in general it will certainly not be so. We must use caution when applying physical constraints on the true sky to our model of it. But the constraints are all we have to guide the algorithm, so in general we will press forward and simply try to decide after the fact if some of our assumptions have been violated.

A particularly useful assumption is bounded support — the sky is mostly empty at radio wavelengths. This is probably the most important distinction between astronomical and conventional image processing. It is a strong regularizing operation on the problem, as demonstrated in a practical sense by the GSP algorithm. (Frequently, this algorithm will diverge when run with insufficient support constraints, yet when sufficient support constraints are given it can produce a high quality result. An example of this is given in Chapter 8.) As well as regularizing¹ the problem to some extent, bounded support also

¹The term “regularization” is used somewhat loosely in this chapter. A formal definition of regularization

makes the problem considerably more tractable computationally when posed as a linear system. In the usual $\mathbf{Ax} = \mathbf{b}$ form, we have one equation for every location where we choose to enforce agreement between the dirty model and the dirty image, N_ℓ locations in total. At each of these locations, we must evaluate a sum of N_C model components. Bounded support, under the slightly questionable assumption that the model will be nonzero in locations where the image is nonzero, means that we do not have to include regions of the model where we believe a priori that there is no emission. In the matrix representation of the linear system, this corresponds to striking out columns. There is also nothing that compels us to use the equations generated by all the pixel locations in the image plane. If we believe that we have enough information, we may also arbitrary drop some of the equations, and choose to work with a subset. In general, more equations would be better, but we must make some concessions to the realities of computation. I call the collection of pixel locations which generate the equations we solve, the data window. This is all the raw data that the deconvolution algorithm will have to work with. The collection of places that we allow the algorithm to place flux is the flux window, and is much the same as any ordinary deconvolution support constraint. The beam matrix that connects the N_{flux} values in the component model vector with the N_{data} values in the dirty image vector is of size $N_{data} \times N_{flux}$.

Until now, I have been deliberately vague about the exact locations of model components and dirty image samples. The linear algebraic approach lends itself well to flexibility in discretization of the problem, and one can easily imagine applications that take advantage of this. One could place more model components in regions where the source varies quickly, for example. The algebraic formalism is also easily generalized to space-variant pixel models and point spread functions, though this is not used in this dissertation. The formalism then begins to converge with the Pixon approach of Puetter and Piña (1994) which does indeed seem a productive line of research. For our purposes here, the full-sized dirty image, dirty beam, and model components are all sampled or spaced at regular intervals on a grid,² $\ell_I i \equiv i\Delta\ell$ and $\ell_C j \equiv j\Delta\ell$. $N_C \equiv N_\ell \equiv N$. The discussion here will use a one-dimensional image for clarity, though the formalism is easily extended to two dimensions as shown by Andrews & Hunt (1977) and others. Inspection of the discretization

for the deconvolution problem is found in Rushforth (1987). A general discussion of regularization at a more generally accessible mathematical level is given in Press et al. (pp. 795–806, 1986). Essentially it means to take a problem which is ill-posed or noisy or both and select a particular solution by the introduction of additional information. Usually the process is controlled by a regularization parameter which determines the relative importance of agreement with the data and agreement with the additional information. The information may be in the form of an extra term in the optimization objective function, in which case the regularization parameter is simply a scalar multiplying the extra term, or it may be defined implicitly via finite iteration of an algorithm, in which case the number of iterations functions as a regularization parameter.

²Again we choose to work in the indexing system where $i = j = 0$ is the center of the image. Index arithmetic is assumed to operate modulo N_ℓ or N_C as appropriate.

equation 2.19 repeated here

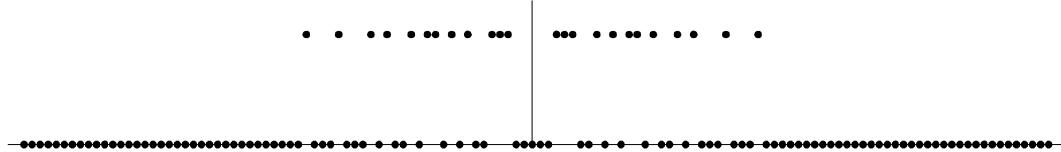
$$(C * B)(\ell_{Ii}) = \sum_{j=1}^{N_C} C_j (P(\ell) * B(\ell))(\ell_{Ii} - \ell_{Cj}), \quad (2.19)$$

shows that for the usual case of $P(\ell) \equiv \delta(\ell)$ the elements of the full sized beam matrix are just $B_{ij} = B((i-j)\Delta\ell)$, which is Toeplitz in form and symmetric due to the symmetry of the beam. When using the discrete Fourier Transform of length N to calculate the images, the symmetry of the beam and periodicity of the transform produce the additional symmetry in the discrete beam $B(i\Delta\ell) = B((N-i+1)\Delta\ell)$.

The periodicity of the transform is sufficient to ensure that the full beam matrix is a circulant, where a circulant is a matrix determined entirely by the first row. Lower rows are simply the right shift by one element of the row directly above. Circulants are desirable matrices in that they can easily be diagonalized by the Fourier Transform matrix. The singular value decomposition of a circulant is similarly trivial to write down. Consequently, many techniques of algebraic image restoration involve ways of manipulating the problem until the beam matrix can be transformed into or approximated by a circulant or block circulant matrix.

That said, here we will do no such manipulations. We wish to encode the a priori support information about the problem into the beam matrix itself. The way to do that is exactly what we described above in the context of the linear equations. We pretend the pixels we do not want in the model do not exist, and strike the columns out of the beam matrix and the corresponding elements out of the model vector. We retain as many equations as we can, preferably in region where there is significant flux, and drop the rest. That is, we strike out rows of the full beam matrix and elements of the dirty image vector. We are right back to where we were before we assumed a regular grid, with an arbitrary sized matrix of no particular symmetry. We have given up the freedom to place flux anywhere we like, to no particular gain in computation. Clearly the flexibility of the approach is not being fully exploited here! But it is still there waiting to be used in future work.

An example of a one-dimensional beam matrix is given in Figure 4.1. (The other panels in the plot will be discussed momentarily.) The one-dimensional sampling which generated this beam is simply



and the corresponding one-dimensional PSF is

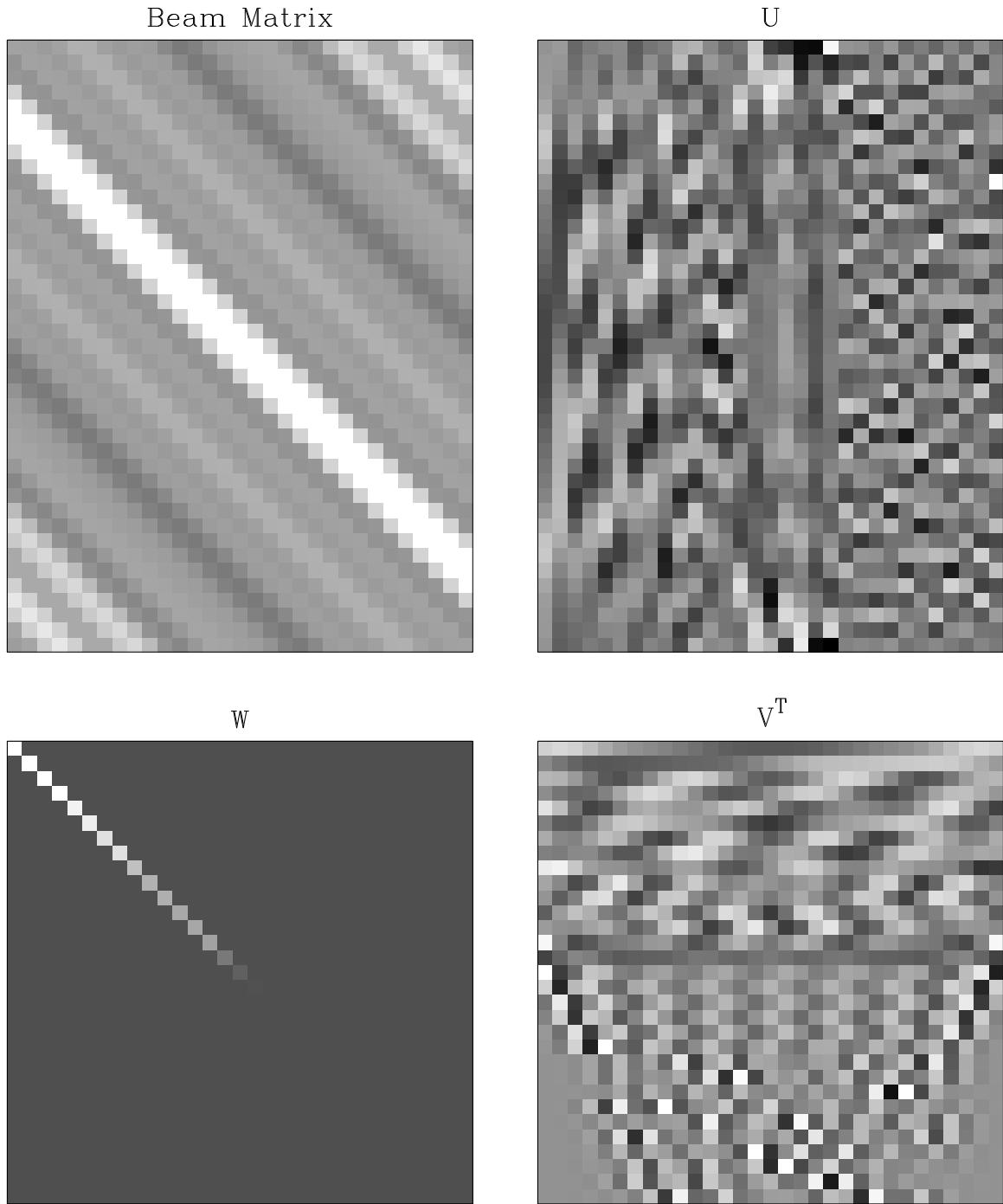
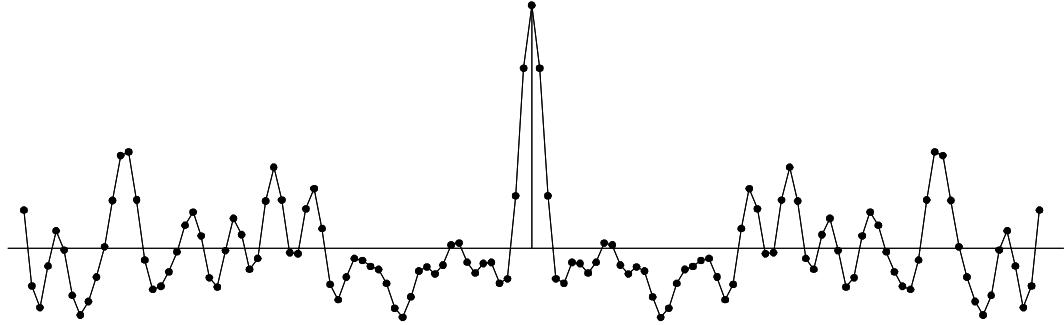


Figure 4.1: Singular Value Decomposition of a one-dimensional PSF



The data window was a 41-pixel-long segment centered at the origin, and the flux window was a 31-pixel-long segment also centered at the origin. In order to work with two-dimensional images in one-dimensional vectors, we merely need a stacking operator. In SDE, there is a subroutine which reads an arbitrary portion of an image into a vector. It simply performs a raster scan of the image, left-to-right, and then top-to-bottom, copying the appropriate pixel to the vector whenever the corresponding pixel in the mask image is true. There is the inverse routine which maps a vector back into a masked image. And there is a routine which takes two masks and a beam image and produces the beam matrix connecting them. This is the only geometric knowledge that the deconvolution program needs. Everything else is encoded in the beam matrix. Figure 4.2 shows what a two-dimensional beam matrix looks like. It is taken from the supernova project in Chapter 8, and maps a 24 pixel diameter circle at the center of the dirty map to another circle of the same size on the component model image. The beam values which fill it are from the uniform weighting trace on Figure 3.19.

We now have a fully realized matrix equation equivalent to the original convolution equation. What do we do with it?

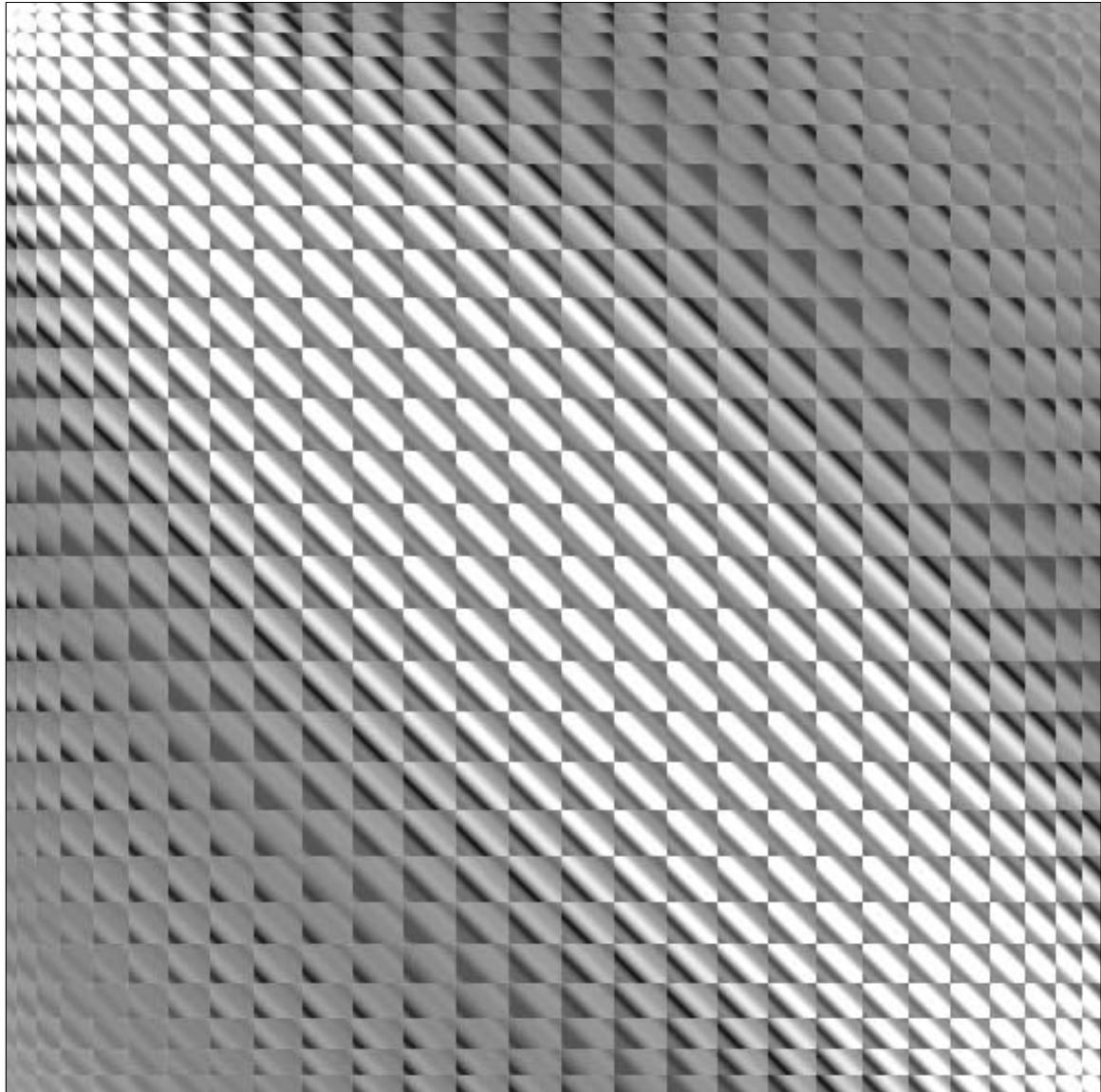


Figure 4.2: Two-Dimensional beam matrix.

4.2 Singular Value Decomposition

An obvious thing one might want to do with a matrix is invert it. Unfortunately, our matrix equation was derived from a singular integral equation, and will usually also be singular. A very general linear algebraic tool for dealing with singular or near-singular matrices is the Singular Value Decomposition, (SVD). It can be shown, (Noble and Daniel, 1977, p. 327), that any $m \times n$ matrix \mathbf{A} with $m \geq n$ can be written as $\mathbf{A} = \mathbf{U} \mathbf{W} \mathbf{V}^T$, where \mathbf{U} and \mathbf{V} are orthogonal and \mathbf{W} is diagonal. (Minor modifications are needed to cover the case $m < n$, but we will not digress here.) \mathbf{U} has the same $m \times n$ dimensionality as \mathbf{A} , (if $m \neq n$, then \mathbf{U} is only column orthogonal), \mathbf{W} and \mathbf{V} are each $n \times n$. The orthogonality property means that $\mathbf{U}^T \mathbf{U} = \mathbf{V} \mathbf{V}^T = \mathbf{I}$. The elements along the diagonal of \mathbf{W} are singular values of \mathbf{A} and the eigenvalues of $\mathbf{A}^T \mathbf{A}$. They are hence guaranteed to be nonnegative. If the matrix \mathbf{A} happens not to be singular, these properties mean that we can immediately write down the solution to the problem $\mathbf{Ax} = \mathbf{U} \mathbf{W} \mathbf{V}^T \mathbf{x} = \mathbf{b}$, namely

$$\mathbf{x} = \mathbf{V} [\text{diag}(1/w_j)] \mathbf{U}^T \mathbf{b} \quad (4.1)$$

This expression makes manifest the dire consequences if one of the singular values happens to be zero and we naively try to solve the equation in this way. If one of the w_j does happen to be zero, it means that \mathbf{A} is singular, and in that event there will either be no solutions to the equation or infinitely many. The SVD has the property, however, that if \mathbf{b} is in the range of \mathbf{A} so that there are infinitely many solutions, we may determine the particular value of \mathbf{x} which has minimum length by replacing all the $1/w_j$ terms in equation 4.1 with 0 when $w_j = 0$, (Press et al., 1986, p. 54). If \mathbf{b} is not in the range of \mathbf{A} , then the prescription above will yield the \mathbf{x} that best minimizes the residuals $\|\mathbf{Ax} - \mathbf{b}\|$. Now it will usually happen that some values w_j will be very small. This corresponds to linear combinations of the singular vectors that are not well constrained by the data. In particular, they will be very sensitive to minor perturbations due to roundoff error or noise. Barring finite arithmetic, the residual norm is guaranteed to go down, the more singular values are included. But the reduction in residual norm caused by adding in negligibly small singular values will likely be tiny, and will be purchased at the cost of a wildly oscillating system of positive and negative values, all delicately cancelling out — if it can even be computed in finite arithmetic, of course! It is often better to discard more singular values than just the zeros. In general, one can use the number of singular values included in the solution for \mathbf{b} as a regularization parameter, trading off agreement with the measured data for the stability of the solution.

Examine Figure 4.1 once again. Notice that the singular values along the diagonal of \mathbf{W} have been arranged in nonincreasing order. The columns of \mathbf{U} and rows of \mathbf{V}^T that correspond to the larger singular values are the smoothest ones. These are the singular vectors of \mathbf{A} and it happens pragmatically that when using the number of singular values as a regularizer, solutions with lower numbers of singular values are smoother. Now examine Figure 4.3. These are singular value spectra as the size of the flux window is varied. The size both of the data window and the dirty map is 128 pixels. A flux window of that

size corresponds to an unwindowed decomposition. In this case, we can actually calculate the SVD decomposition analytically. One of the pleasant properties of circulants is that their singular values are the DFT of their top row, (Hunt, 1971). In the case of a dirty beam, the singular values are the gridded weights. This PSF was created with uniform gridded weights, so the singular values of the full PSF should be constant or zero. The entire long shelf of low values are spurious artifacts introduced in the decomposition. For numerical stability reasons alone, one must trim at a singular value threshold greater than 10^{-6} of the peak to remove these artifacts. Of course, after trimming away the spurious singular values for the unwindowed case, one simply re-obtains the original gridded weights. Multiplying by the singular vectors of a circulant is a DFT, so one gets back the uniform dirty map. (Even if the original weighting was not uniform, the gridded weights are divided by themselves, and one gets back the uniform dirty map.) If one throws away more singular values than the zeroes in the unwindowed case, one merely obtains another dirty map with fewer measurements. Clearly, in order to generate unmeasured spatial frequencies, one must encode some support information into the structure of the beam matrix. I do not know of a quantitative measure of how much information is included by tightening the support constraints, but it is worth noting that the fraction of significant singular values above the obvious cutoff grows as N_{flux} is decreased. I can say from a pragmatic standpoint that a window of 20 pixels out of 128 was sufficient to give reconstruction on a par or better than CLEAN.

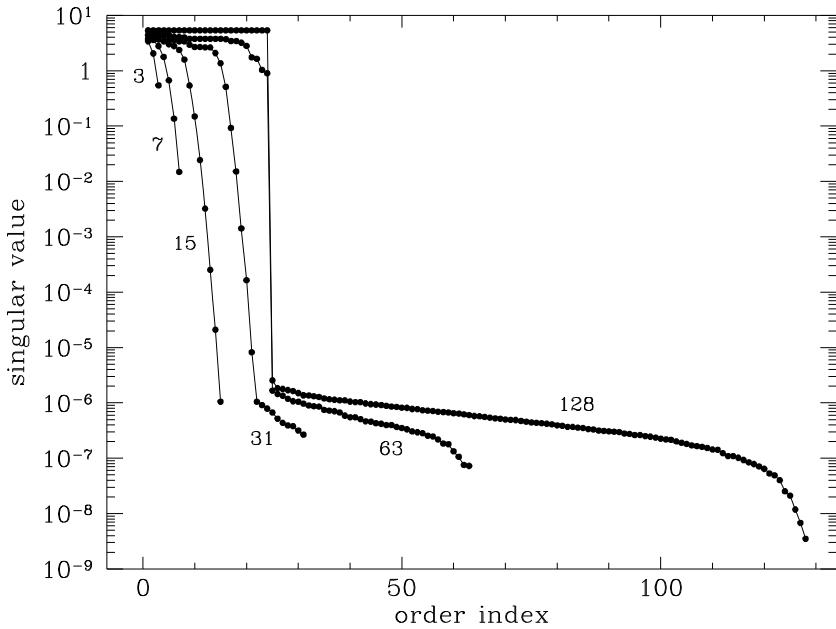


Figure 4.3: Singular value spectrum against window size.

The computational demands of direct algebraic methods are considerable. Several hours are required on an IBM RS/6000-580 workstation for an SVD deconvolution with

$N_{flux} = N_{data} = 3000$. The runtime scales with $N_{data} N_{flux}^2$, so there is a strong dependence on problem size. An even worse problem is memory, since that effectively sets a hard limit on the size of the problems which can be solved. At the least, one must hold a copy of the beam matrix in memory, of size $4N_{flux} N_{data}$ bytes, for normal real precision arithmetic. The algorithms examined for the SVD are not optimized for sequential memory access. If the beam matrix must be swapped to disk, the efficiency drops by a large factor, and problem which were just barely practical by runtime become completely unfeasible. SVD is even worse than other algebraic methods, in that it requires both the **U** and **V** matrices to be in memory at the same time. If the data and flux windows are the same size, this doubles the required memory over algorithms like NNLS which operate with a single copy of the beam matrix. Two different implementations of the SVD were examined, **svdcmp** from Numerical Recipes, (Press et al., 1986) and **SGESVD** from LAPACK, (Anderson et al., 1992). While the LAPACK routine was somewhat more robust than the Numerical Recipes routine, (the NR routine would occasionally fail to converge), the solutions produced by both were quite comparable. A more telling difference was between the single and double precision versions of these routines, and the double precision solutions were clearly superior.

I do not wish to belabor the SVD overlong, since by computational requirements and quality of solution both, it is inferior to NNLS. But it has been worth mentioning as a direct demonstration of how support information generates unmeasured spatial frequencies. And it can work reasonably well, if a human is examining the output and exercising judgement about where to make the singular value cutoff. Closing this section, I show two model deconvolutions of the usual uniformly-weighted 3C48 Gaussian model source. Comparing these reconstructions to Figure 6.11, we find that the single precision deconvolution is slightly worse than the CLEAN reconstruction, and the double precision deconvolution slightly better. Certainly the improvement in solution is not worth the dramatic increase in computation. The SVD by itself is interesting, but not worth pursuing unless it can be combined with another algorithm that needs the stabilizing properties of the SVD.

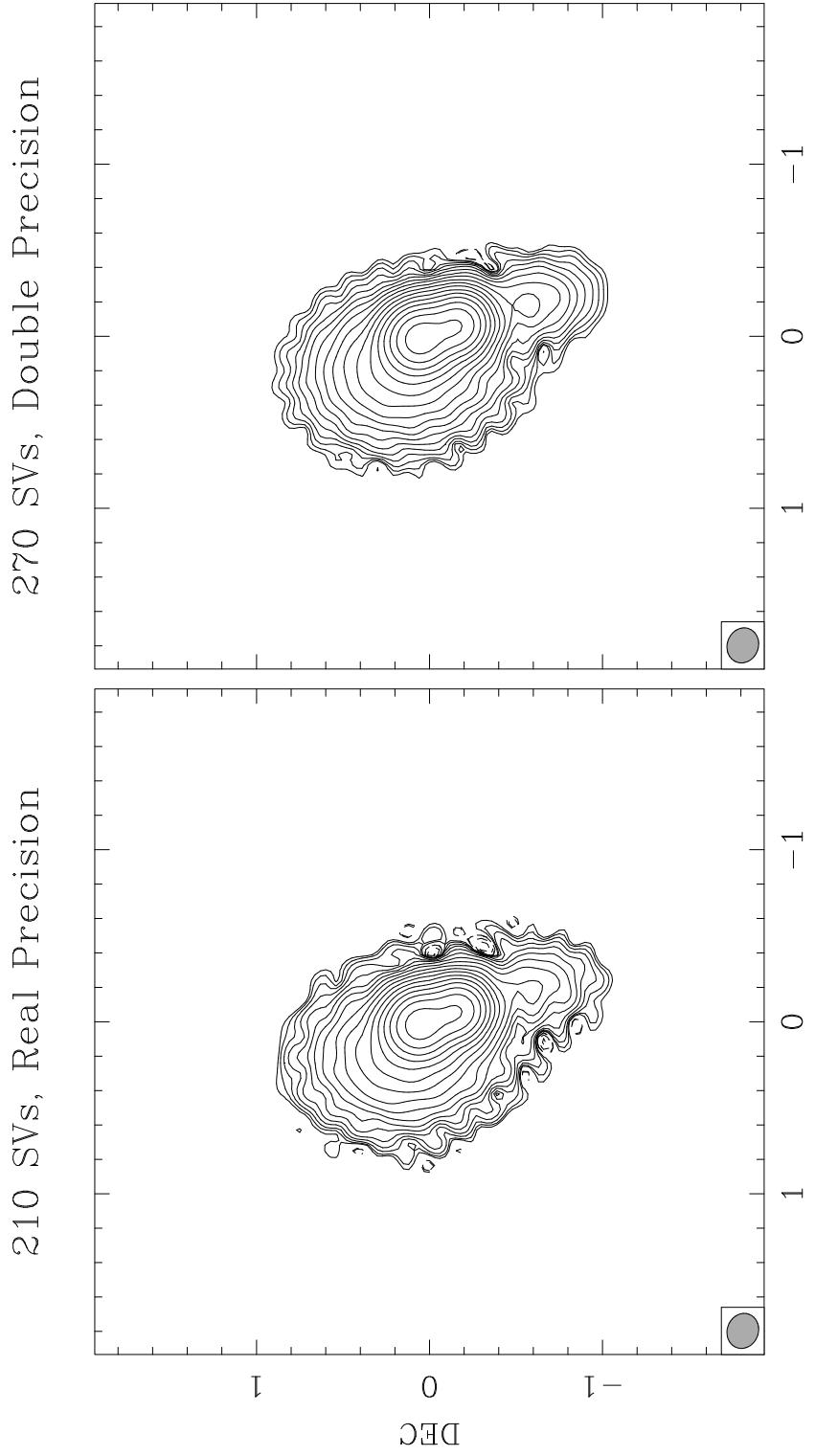


Figure 4.4: SVD deconvolutions of 3C48 model source. The left panel used the Numerical Recipes routine, `svdcmp` in real precision with 210 singular values included in the solution. The right panel used the double precision LAPACK routine `DGESVD` and included 270 singular values. Both use the manually selected optimum number of singular values for the reconstruction. Lowest contour is 100 μ Jy/beam, and there are 4 logarithmic contours per decade.

4.3 NNLS

SVD deconvolution happened to be the original motivation for the algebraic investigation. But once the basic framework was in place, it was a simple matter to try another algorithm which also purported to solve the basic matrix equation. The algorithm, NNLS from Lawson & Hanson (1974), also had the considerable advantage of enforcing positivity on the solution. This experiment was performed while working on a project to image the recent supernova SN1987A, and it was naturally applied there first. The project had a low signal-to-noise ratio, and was relatively simple to deconvolve at the nominal resolution. NNLS, like most other algorithms, produced an acceptable solution. The slipping point was that the project required the highest possible resolution and hence direct examination of the deconvolved model components. As will be shown by example later, the NNLS components do not appear terribly physical when examined directly. They tend to be spatially isolated from each other and spiky to the eye. CLEAN shares this behavior to an extent, and neither CLEAN nor NNLS was deemed suitable for the needs of the SN1987A project. Due to other concerns, the implementation was largely forgotten until the time came to write up this chapter as “an interesting exercise but ultimately not a particularly useful one.” At this point, I resurrected the program and finally tried it on the model 3C48 source which had given so many other algorithms difficulties; I was shocked at how well it did! After subjecting the algorithm to the same suite of algorithmic test cases as the others, or at least as many of them as could be accommodated within the computational limit, I was eventually convinced that it really was performing as well as the initial tests had indicated — a significant advance in high dynamic-range imaging. The simulations did indicate that the NNLS solution degrades with source size compared to other algorithms, MEM in particular, and that the crossover point where it became comparable to other algorithms was somewhere near the limit of what could be processed computationally. The size scales at which NNLS excels are sources which are too compact for MEM to do a good job, and which are too extended for CLEAN to process well. Since the latter limit is surprisingly small, there are a significant number of scientifically interesting sources for which NNLS or a derivative algorithm is the tool of choice.

In an especially apt case of timing, the NRAO Very Long Baseline Array was just being brought up to full capacity. For the first time, a VLBI array was producing observations theoretically capable of being imaged at many tens of thousands to one dynamic range. And the images being produced were suffering exactly the same CLEAN artifacts as demonstrated in Chapters 5 and 6! NNLS deconvolution was attempted on the test observation of the bright compact source DA193, and it worked quite well as shown in Chapter 9. In the process, another useful property of NNLS deconvolution was discovered — it works particularly well as part of the hybrid mapping loop, where the data are iteratively imaged and self-calibrated. It was found that the interaction of CLEAN and self-calibration was limiting the composite process to somewhat lower dynamic range than CLEAN was capable of on perfectly calibrated data. In contrast, NNLS has consistently produced images of simple sources down to the thermal noise limit.

4.3.1 The NNLS algorithm

The reference for all of this section is Chapter 23 of Lawson & Hanson (1974). It provides not only the theory, but also the **FORTRAN** code that is the heart of the deconvolution program. I will describe the working of the algorithm in general terms, but it is both redundant and beyond the scope of the dissertation to do more than that.

The general class of problems to which NNLS belongs is the least squares problem with linear inequality constraints. That is, let \mathbf{E} be a $m_2 \times n$ matrix representing the experimental design or deconvolution problem, \mathbf{f} be the m_2 element data vector, \mathbf{G} be an $m \times n$ constraint matrix and \mathbf{h} be an m element constraint vector. The basic problem to be solved, Problem LSI, is

$$\text{Minimize } \|\mathbf{Ex} - \mathbf{f}\| \text{ subject to } \mathbf{Gx} \geq \mathbf{h}.$$

For our purposes here, \mathbf{G} is the identity matrix, and \mathbf{h} is identically zero. There is a condition which all solutions of the problem LSI must satisfy, called the Kuhn-Tucker conditions. Conversely, any vector which satisfies them is a solution. First note that the constraint equations can be interpreted as a series of m simple linear-inequality constraints. Every row of the constraint equation, $\mathbf{g}_i^T \mathbf{x} \geq h_i$ divides the allowed solution space in half with a bounding hyperplane. A given solution vector \mathbf{x} can be either interior to, or on the bounding hyperplane for each row. The Kuhn-Tucker conditions say that the negative gradient vector of $\phi(x) \equiv \frac{1}{2} \|\mathbf{Ex} - \mathbf{f}\|^2$ points into the cone spanned by the negative normals to the constraint hypersurfaces containing \mathbf{x} . This means that in some appropriate coordinate system, the solution must have zero gradient in the directions where the solution is not hard up against a constraint surface. In the directions where it is on the surfaces, the negative gradient must point into the surfaces, not away from them.

It may help to visualize the case in Cartesian 3-space, where we wish to minimize $\|(x, y, z) - (0, -1, 1)\|$, while satisfying $y > x + 1$, $y > -x + 1$, and $z < 10$. The first two intersecting planes form an edge at $(x, y) = (0, 1)$, and clearly the point on that edge nearest $(0, -1, 1)$ will be the solution, namely the point $(0, 1, 1)$. The constraint that $z < 10$ is not coming into play, so there had better be no gradient projection along the z axis, or the solution cannot be a global minimum. The negative gradient is in fact $(0, -2, 0)$, which is normal to neither constraint plane, but “against” the sense of both.

The actual NNLS algorithm holds two index sets of indices, (or rather, one set and its complement), to indicate whether a given tentative solution element is zero (on a constraint surface) or not. Elements whose index is in the set \mathcal{Z} are held at the value zero. Elements whose index is in the set \mathcal{P} are free to take arbitrary values.

1. Set all elements of the solution vector \mathbf{x} to zero, set all of the indices into the set \mathcal{Z} , set \mathcal{P} to the empty set.
2. The main loop begins here. Compute the gradient vector \mathbf{w} from the current value of \mathbf{x} : $\mathbf{w} = \mathbf{E}^T(\mathbf{f} - \mathbf{Ex})$

3. If \mathcal{Z} is empty or if all elements of \mathbf{w} with indices in \mathcal{Z} have values ≤ 0 , we have a solution. Terminate the algorithm.
4. Find the maximum element of \mathbf{w} , which will be positive. Move its index from the set \mathcal{Z} to the set \mathcal{P} .
5. A possible secondary loop begins here. Form a modified version of the \mathbf{E} matrix, \mathbf{E}_{pos} , where the columns corresponding to indices in \mathcal{Z} are replaced with columns of zeros. Solve the unconstrained least squares problem $\mathbf{E}_{pos}\mathbf{z} = \mathbf{f}$. This will only determine the components of \mathbf{z} corresponding to indices in \mathcal{P} . Set the remaining elements of \mathbf{z} to 0.
6. If all the elements of \mathbf{z} with indices in \mathcal{P} are in fact > 0 , this is an acceptable new trial solution. Set $\mathbf{x} = \mathbf{z}$ and continue with the main loop at (2).
7. We can only accept a fraction of \mathbf{z} as the new trial solution: Find an index q such that $x_q/(x_q - z_q)$ is the minimum of all such expressions for negative elements in \mathbf{z} . For this q , call the expression α .
8. Form the linear sum $\mathbf{x} = \mathbf{x} + \alpha(\mathbf{z} - \mathbf{x})$
9. Move from the set \mathcal{P} to the set \mathcal{Z} all indices for which the corresponding element of \mathbf{x} is zero. This will include q , and may include other elements as well. Continue the secondary loop at (5)

The computation of the subset unconstrained problem in step (5) is not actually calculated by filling unused columns with zeros, of course, but rather with an embedded QR decomposition that knows not to access elements beyond that allowed by the index sets. Also adding greatly to the efficiency of the implementation is that at each pass through step (5), we are solving nearly the same problem as before. The QR decomposition of \mathbf{E}_{pos} is retained across iterations, and updated whenever an index changes sets. (Methods exist to update a QR decomposition in much less time than computing the entire decomposition from scratch.)

It is proved in Lawson & Hanson that the iteration of the NNLS algorithm is finite. Given sufficient time, the algorithm will reach a point where the Kuhn-Tucker conditions are satisfied, and it will terminate. There is no arbitrary cutoff in iteration required; in that sense it is a direct algorithm. It is not direct in the sense that the upper limit on the possible number of iterations that the algorithm might need to reach the point of optimum solution is impossibly large. There is no good way of telling exactly how many iterations it will require in a practical sense. The solution does improve smoothly as the iteration continues. If it is terminated early, one will obtain a sub-optimal but likely still fairly good image. Defining an iteration as the number of times the algorithm passes through step (5), Lawson & Hanson comment that their problems typically terminated after $n/2 = N_{flux}/2$

iterations. They hard coded a limit of $3n$ iterations into their code. I found it occasionally necessary to exceed that limit in model testing, and have allowed the maximum iteration number to be a free parameter. Certainly the algorithm is normally allowed to terminate on its own. The number of iterations required is also a function of the signal-to-noise ratio in the data. Noisy data means that the algorithm will try to fit negative flux to the noise power and will more quickly reach a situation where it cannot improve the solution by adding more components. The longest convergence times seen were roughly $10N_{flux}$, on large model objects that had been noiselessly sampled in the sky plane prior to creation of the dirty map. That is the, the most perfect possible model data set required several times Lawson & Hanson's hard coded maximum. All observational data required many fewer iterations, and none exceeded the original limit of $N_{flux}/2$.

4.3.2 How Important are the Windows?

This has not been investigated systematically on model data, but a reasonable amount of experience has been gained with the DA193 and 3C48 projects. It appears that the most critical factor is the size of the source itself, not the size of the window. On the DA193 project, both data and flux windows were varied by a wide range without significantly affecting the result. For the very highest precision deconvolutions, however, the window was edited by hand to exclude the most obvious spurious features at the window's edge. This made a small but noticeable difference in the solution. It would seem that a reasonable strategy is to deconvolve with as large a flux window as one can conveniently compute, using a data window at least as large as the flux window and larger if possible. Deconvolve once or as many times as needed if self calibrating, until one has a good calibration and reasonable image. At this point, direct examination of the component model image will likely show the main source and possibly a slight scattering of very low level components off the main source. The latter can be excluded by hand for the highest quality image. It cannot be said that this procedure is excluding the existence of low emission at a level below that found by the wider windows, but one can claim that the image is consistent with no emission at the lowest level found by any means at all. (A fundamental principle of VLBI imaging has always been "If you see a scientifically interesting feature in your image, try hard to make it go away. If it does go away, it wasn't mandated by the data and you couldn't have trusted it anyway.") In comparisons on DA913, NNLS generally exhibited less sensitivity to support constraints than CLEAN, but was not completely immune to the problem.

4.3.3 Model Deconvolutions

Once again, we return to the 3C48 model source. Since this is shown several places elsewhere with uniform weighting, it was decided to use an intermediate robustness here. This was primarily to provide an example of the robust beams. Granted, this decision does

favor NNLS somewhat, in that NNLS works particularly well with a robust beam, while CLEAN would prefer a uniform beam, but perhaps compensating in the other direction, we have used a model simulated in the visibility plane, and one that has significant image plane representation problems. CLEAN has reached the maximum fidelity it can on this source — issues of representation are not at all important for CLEAN here. A similar deconvolution where the model was discretized in the sky plane prior to creation of the dirty image yielded a nearly identical CLEAN image, apart from effects very localized to the region of the worst representation problems. By contrast, NNLS can go an additional factor of 10 deeper in off-source RMS when provided with such an ideal situation. So in selecting the parameters for this simulation, we have made one arbitrary decision that will favor NNLS and one that will favor CLEAN. As will be seen, the results are dramatic enough that it really makes very little difference how the details of the simulation are set up.

First, we show the raw and smoothed model in Figure 4.5. This is the same 3C48 model as used in the SVD section and the weighting chapter. Model parameters are given in Table 3.8, except that the magnitudes of the four outlier point sources have been adjusted to $1.19 \mu\text{Jy}$, $2.56 \mu\text{Jy}$, $5.51 \mu\text{Jy}$, and $11.9 \mu\text{Jy}$. The ratio between adjacent point sources corresponds to one contour in the following plots, the 3rd root of 10 ≈ 2.15 .

Two support windows were used. The ‘Large Window’ is exactly that used in the CLEAN simulations presented in Section 3.10. It contains a fairly generous region for the central flux of 3C48, and circles of 11-pixel radius centered on each offset point source. This box is shown as a greyscale superimposed on the model plots in Figure 4.5. The ‘Tight Window’ is the same, but without provision for the point sources. The idea is that the astronomer would initially box the region of the visible source, and have no idea that the extremely faint sources are there. If the faint sources show up in the residuals, then they can be properly boxed and the deconvolution run again. The current computational limits to NNLS are enough for many practical problems, (roughly 6000 pixels of emission can be processed in a few hours on our IBM RS/6000-580 workstations), but they are not sufficient to allow boxing large regions on the sky just to see what flux might be there. If a faint source is to be detected, it must be initially detectable in the residuals. The pixel and dirty image sizes are the same as before, at $.03''$ and 512^2 , respectively. The CLEAN used 100,000 components at a loop gain of 0.1, which was sufficient to reduce the residuals to $12.2 \mu\text{Jy}/\text{beam}$ in all cases. Except where noted, all contours in this sequence are at the same fixed levels.

Model
Smoothed Model

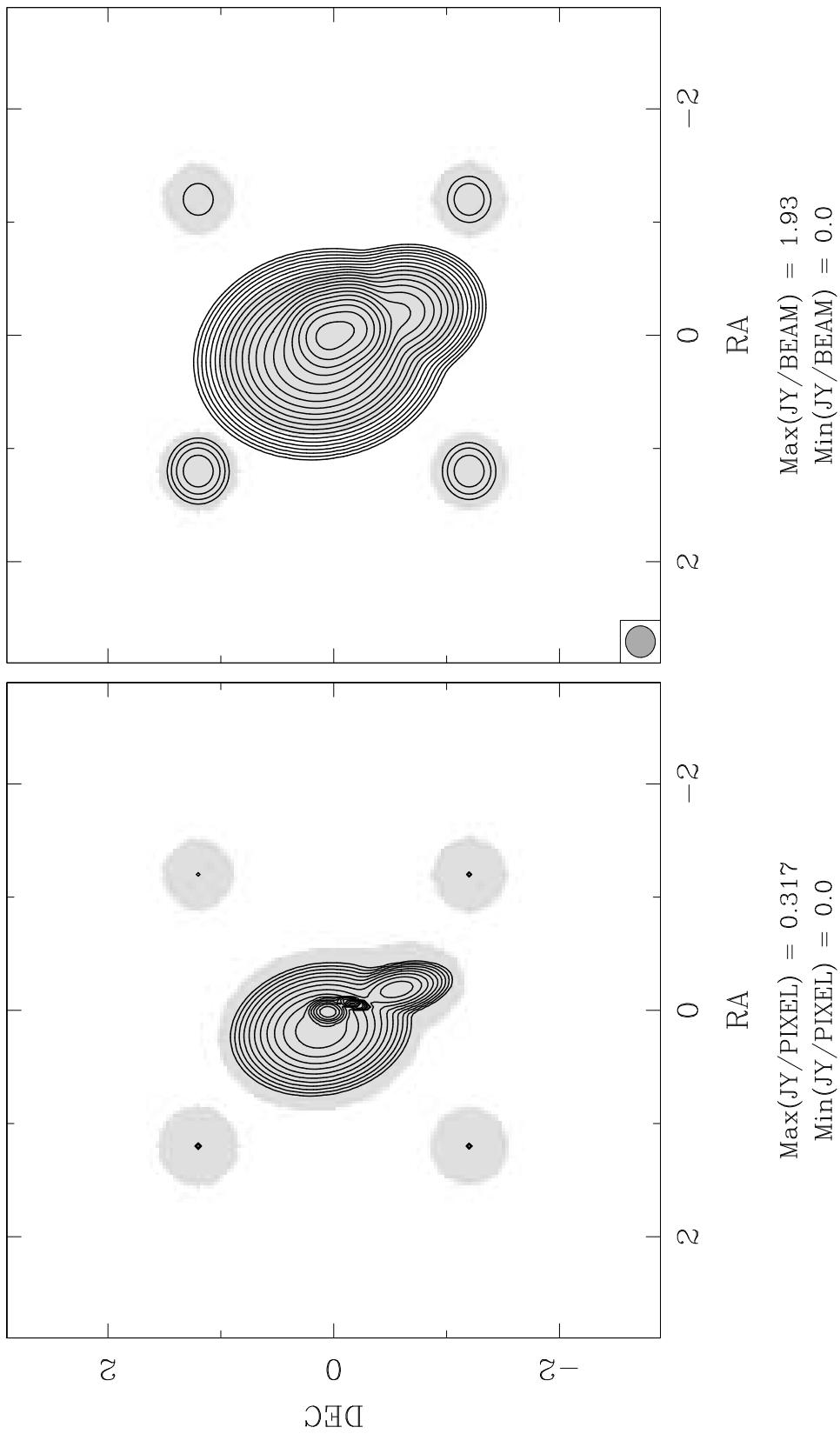


Figure 4.5: Raw and smoothed model source. The widows are the shaded region. The tight window is just the central shaded region, the full window is all grey regions collectively. Three logarithmic contours per decade. Lowest contour is .59 $\mu\text{Jy}/\text{beam}$.

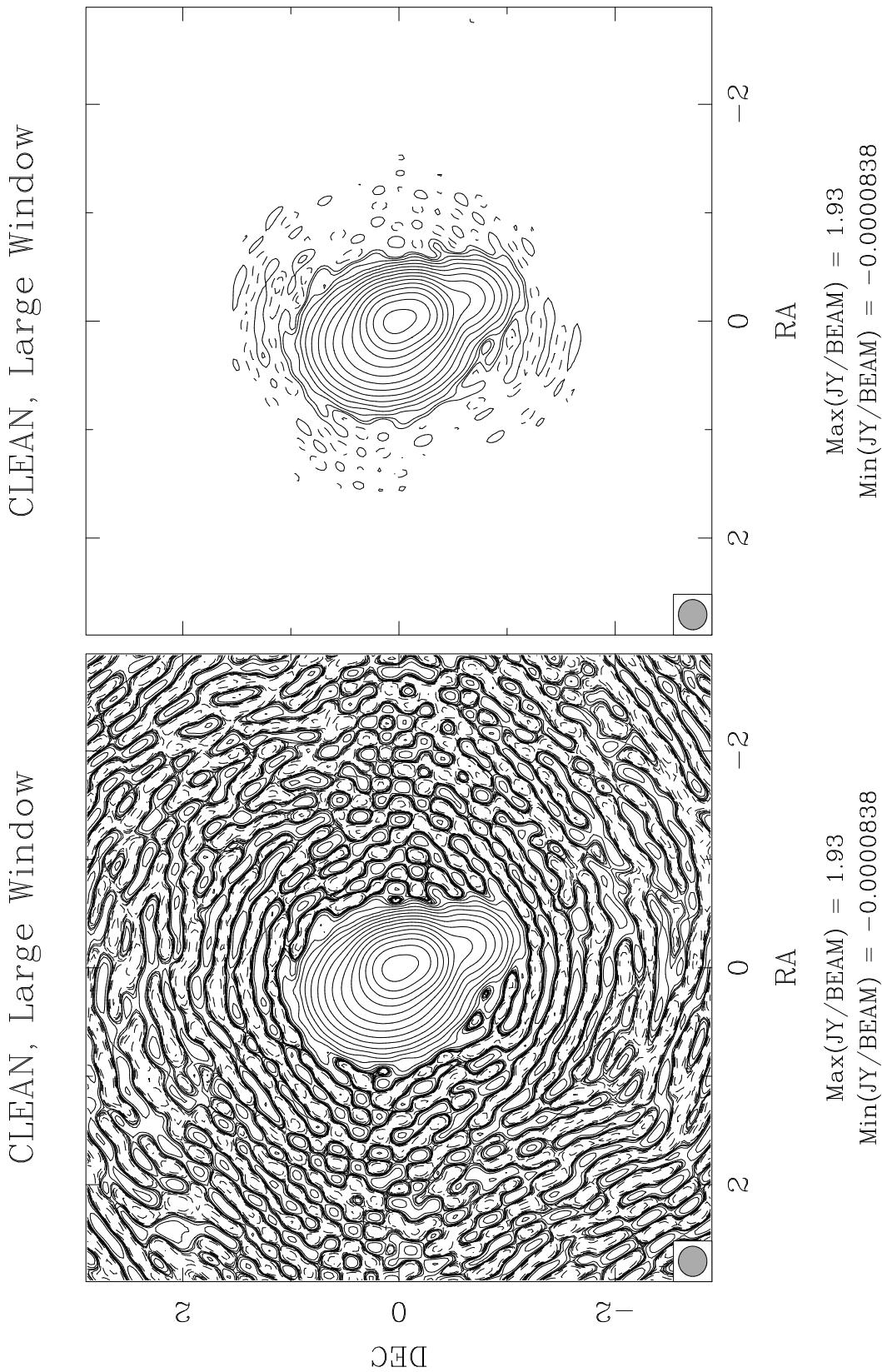


Figure 4.6: CLEAN deconvolution. Noiseless data, modelled in the visibility plane. The right panel is the same image, contoured up by a factor of 50. Lowest contours are $.59 \mu\text{Jy}/\text{beam}$ and $29.9 \mu\text{Jy}/\text{beam}$.

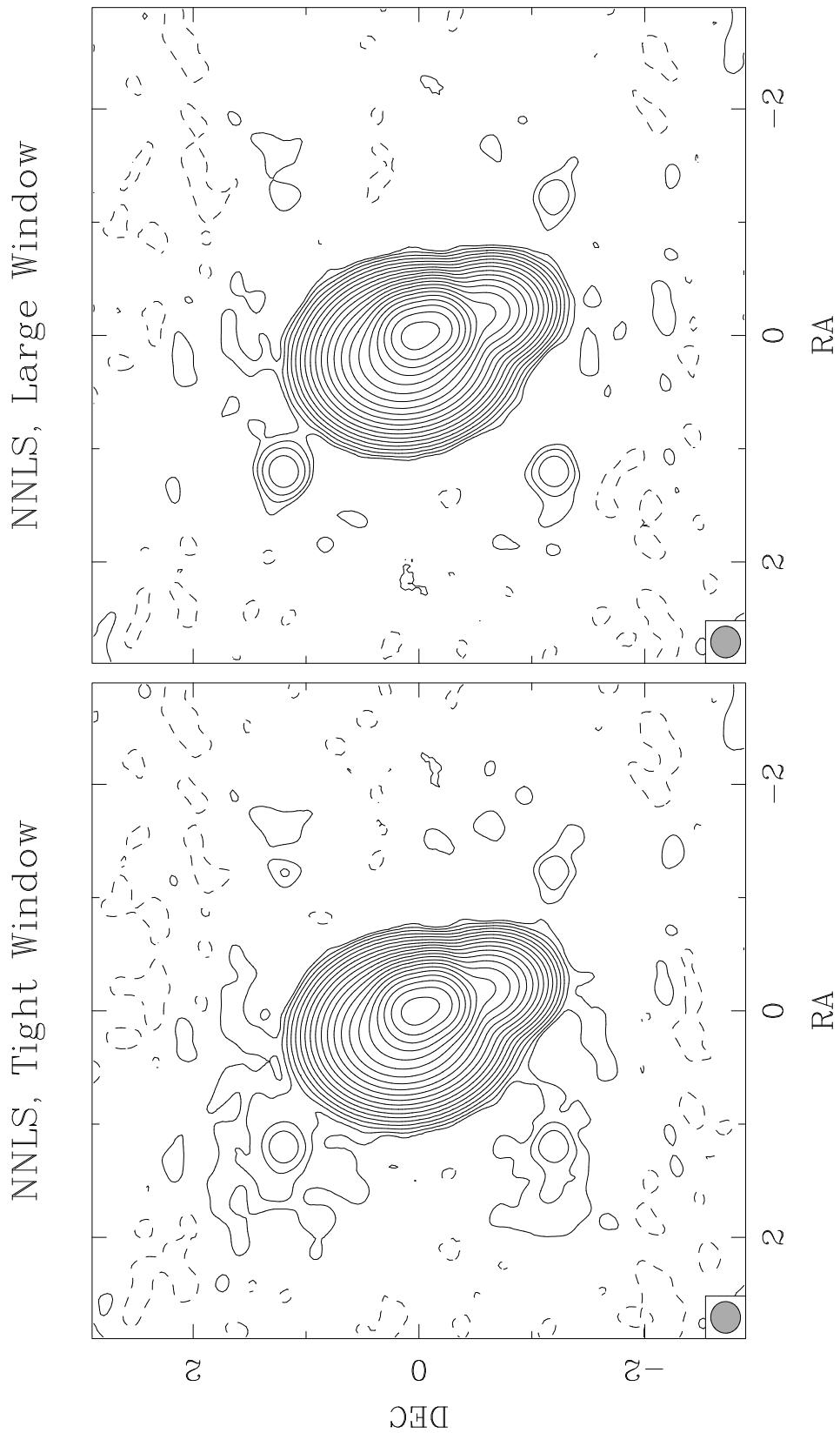


Figure 4.7: NNLSS deconvolutions. The point sources are entering the image via the residuals in the left panel and via the components in the right panel. Lowest contours are .59 μ Jy/beam.

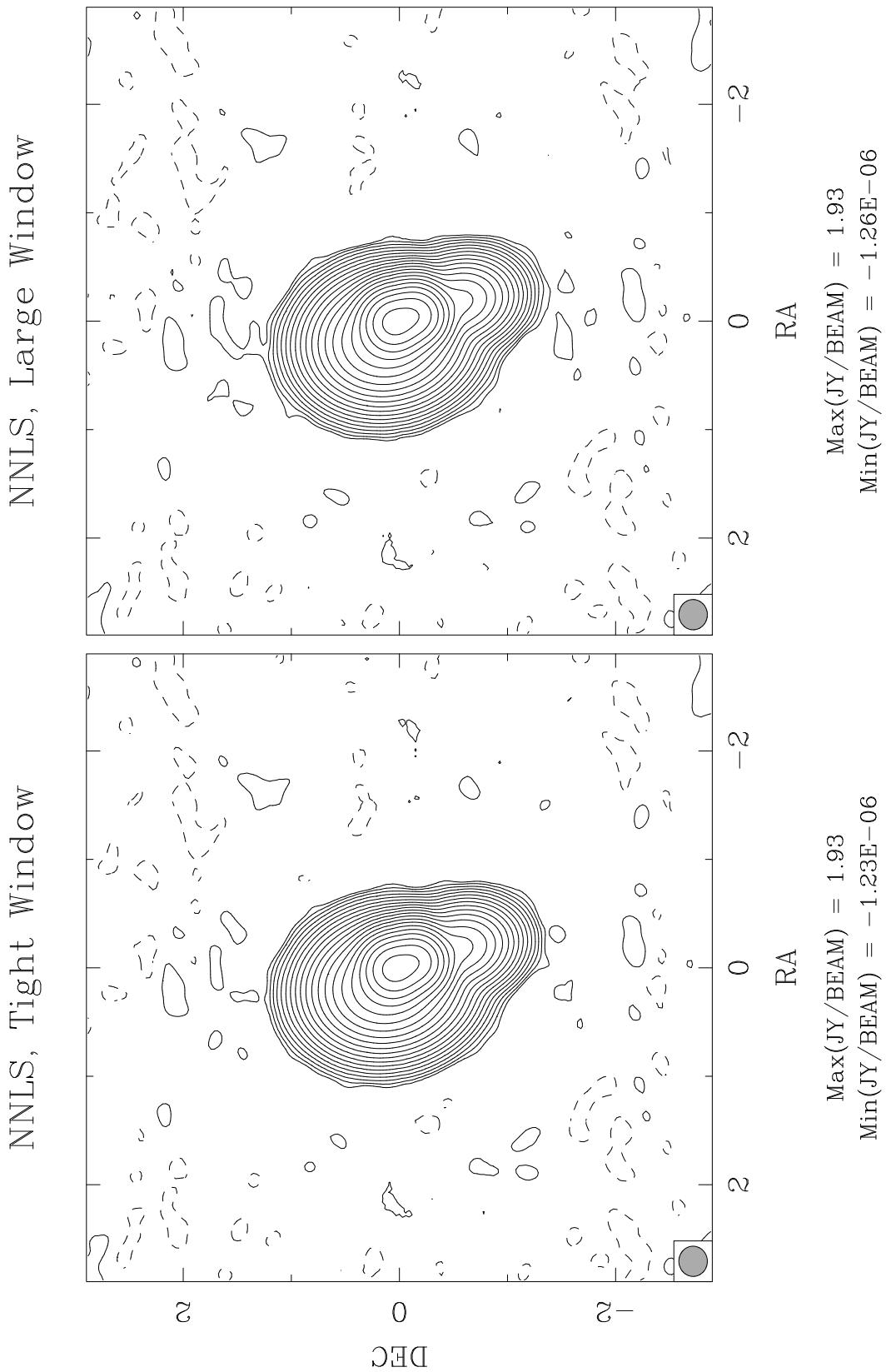


Figure 4.8: NNLS deconvolutions when the point sources have been removed from the model. Simply placing windows on empty sky is not sufficient to create a source.

Next we present the error characterization plots against weighting for NNLS. These parallel the similar plots for CLEAN and MEM in Figures 3.31 and 5.21, respectively. The simulation is identical to that described in Section 3.10, differing only by using NNLS instead of CLEAN. There are fewer points on the noiseless curve than in the other two figures, because the noiseless NNLS deconvolutions were quite computationally intensive, and only enough points to get the general shape of the curve were computed. Things to note about these plots are:

- The worse on-source error for NNLS is nearly independent of weighting in the noiseless case.
- NNLS has the strongest dependence of any of the major algorithms of full-image maximum error against weighting, with a fitted restoring beam. There must be a significant reconstruction error just outside the u - v sampling. CLEAN shows a similar qualitative behavior, if not as strongly.
- There is no significant difference between NNLS, CLEAN, and MEM on the noisy off-source maximum and RMS error cases. They all do pretty well.
- For the noiseless case, the difference between NNLS and the other algorithms becomes more apparent moving from maximum full-image to maximum off-source error, and still more so in terms of the off-source RMS error.
- The merging of the two lines on the noisy maximum off-source error plot indicates only that the worst off source error is a thermal residual that is not being smoothed, (and thus has no dependence on restoring beam). The curves converge on the noisy full-image maximum error plot when NNLS has dropped the on-source error low enough that the worst error is the same off source peak.
- In the noiseless case, the off-source RMS error has a minimum in robustness, right about where the worst near-in sidelobes of the beam have been minimized. While it is very well defined, it is not terribly significant in a quantitative sense, with the extremes differing by about 25%.
- It's interesting that the minimum in off-source RMS is not the uniform beam, which generally has the best far-off sidelobes of any weighting. CLEAN obviously prefers the far uniform beam in the noiseless case, and nearly has a maximum near the same robustness value.
- In the discrete model runs not shown here, the noiseless off-source RMS error for CLEAN was basically the same as in Figure 3.31. CLEAN had already bottomed out due to the algorithm and slight under-CLEANing; the details of the sampling made little difference. The same thing is nearly true for MEM. With the perfect sampling, NNLS reduced the off-source RMS error to of order $2.5(10)^{-8}$ Jy/beam, roughly 10^3 times deeper than CLEAN, and 10^4 times deeper than MEM!

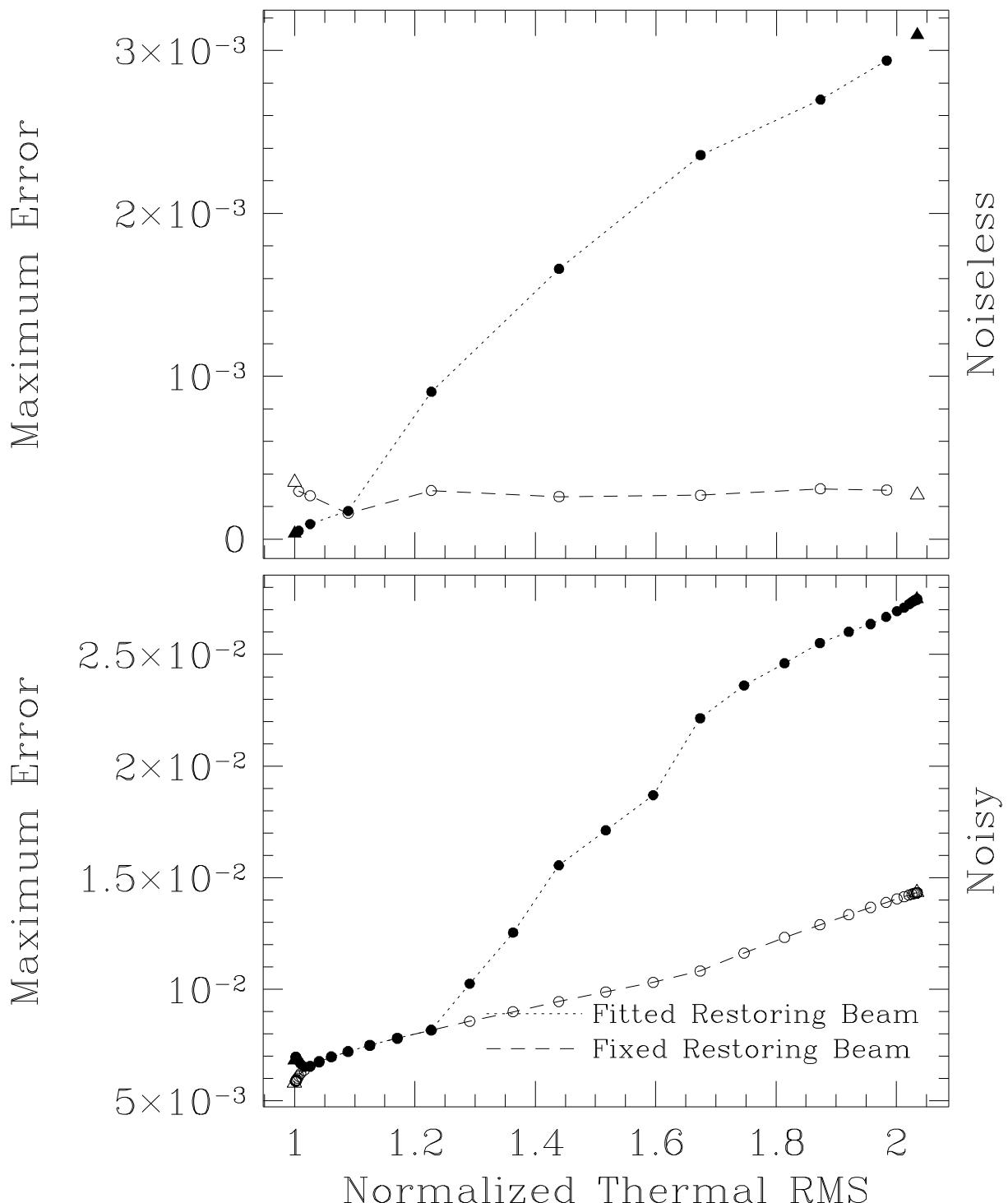


Figure 4.9a: Error quantities against RMS degradation factor: maximum error. The fitted restoring beam corresponds to Figure 3.30. The fixed beam curve is more representative of the absolute deconvolution quality and used $\sqrt{B_{nat}B_{uni}} = .2214''$.

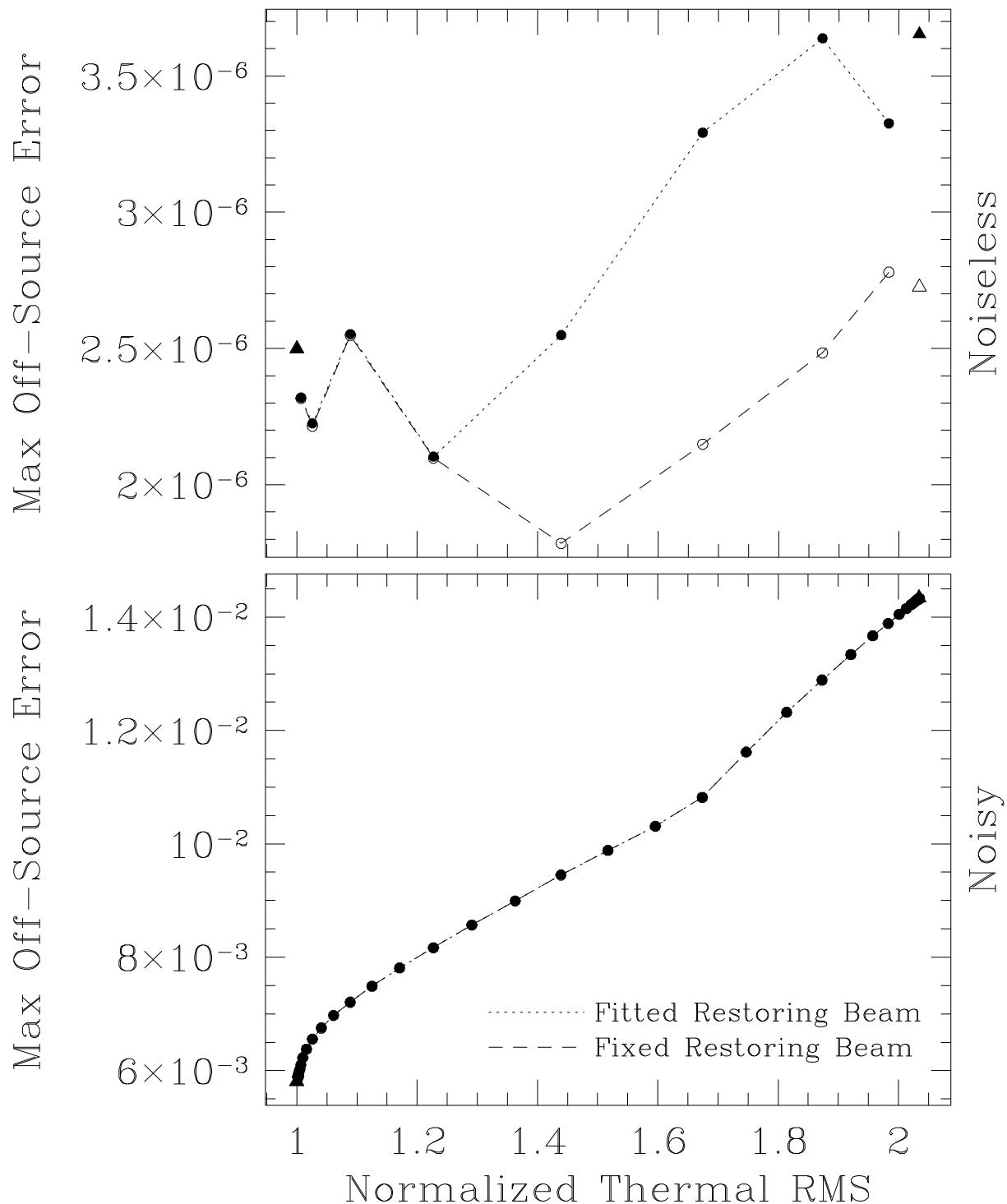


Figure 4.9b: Error quantities against RMS degradation factor: maximum off-source error. The two traces in the lower panel are identical.

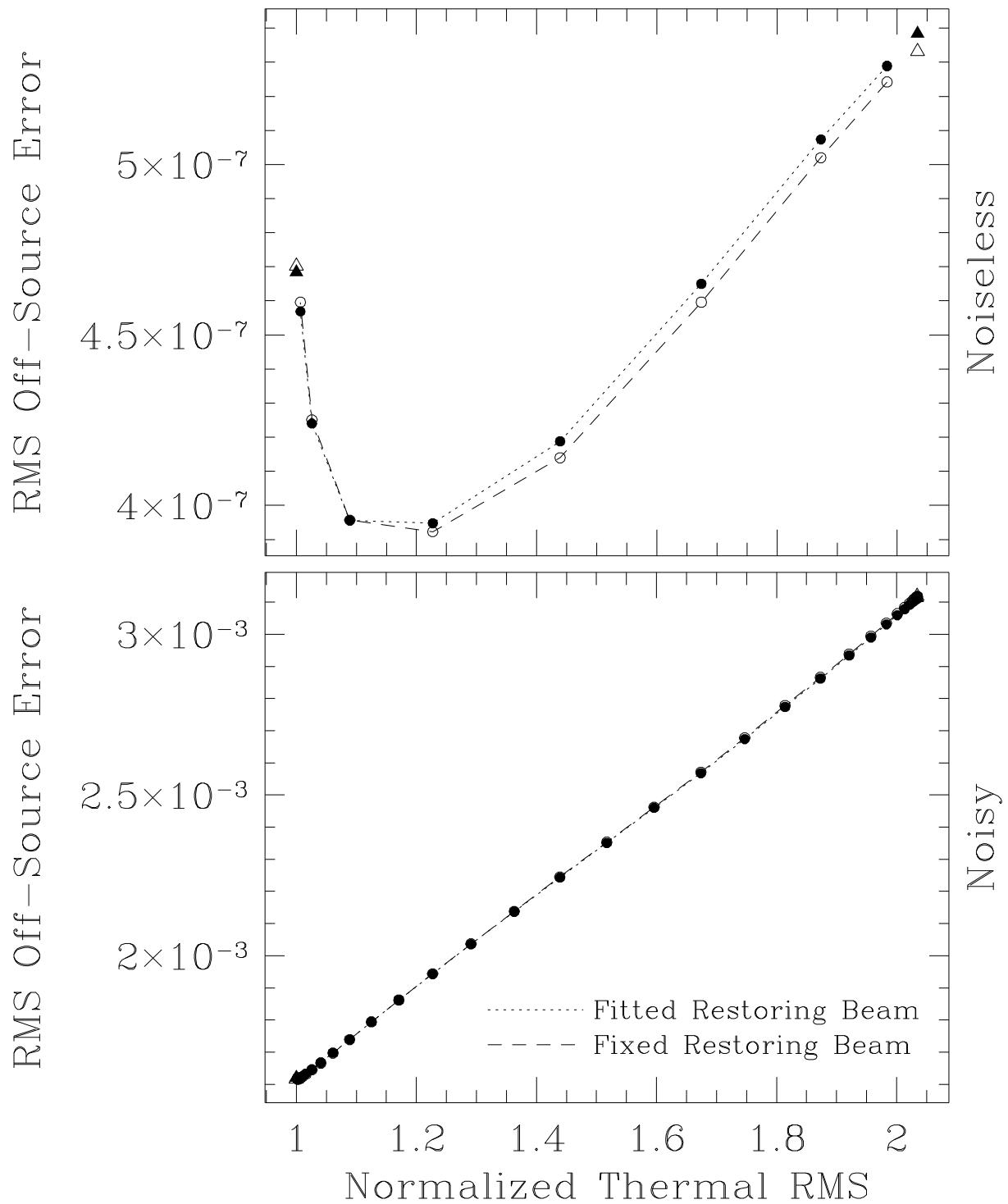


Figure 4.9c: Error quantities against RMS degradation factor: off-source RMS error.

Finally, we present one fairly amusing example both of the flexibility of the algebraic approach and of the robust nature of NNLS deconvolution. At an early stage in the image processing of DA193, the calibration was such that there was a slight extended halo around the source, which later turned out not to be real. For the moment we may take it as the truth image, since that is what is present in the data as far as the deconvolution algorithm is concerned. In fact, such a “truth image” is slightly unfair to NNLS, in that the image it is trying to recover has negative values which may partially disrupt the solution process.

Figure 4.10 shows two deconvolutions of this data set. In the first, the data window is the size of the large grey circle, with a radius of 38 pixels and an area of roughly 4500 square pixels. The flux window is the smaller grey circle, with radius 20 pixels and an area of 1320 square pixels. The left panel is the standard NNLS deconvolution using these windows, and for this demonstration may be considered what the source looks like. In the right panel, the flux window was unchanged; the algorithm could place flux in the proper location. The data window, however, was changed to be the annulus between the two circles. The windows are now completely disjoint. The algorithm was not given the dirty image at the locations where it was placing flux. It reconstructed the source entirely through the sidelobes. As can be seen by comparing features, the second deconvolution is not as good as the first, but it is remarkably close. NNLS is even quite capable of reasonable deconvolutions when there are more pixels in the flux window than in the data window. Obviously, this can only be sensible with the addition information introduced via the positivity constraint. I know of no way to quantify how much additional information is introduced, but in practice it seems that given a significant piece of the dirty image to work upon, NNLS will return a reasonable answer.

I have been calling this a “deconvolution party trick,” since I can think of no good application of this ability in radio interferometry. Locating a far source by its sidelobes leaps to mind, but unfortunately the NNLS algorithm cannot image a large field without knowing where to place the support windows. In the optical or any image plane detection system, of course, one may have the problem of missing data due to detector saturation or to cosmic rays. This has not been pursued at all, but it seems like it could be useful.

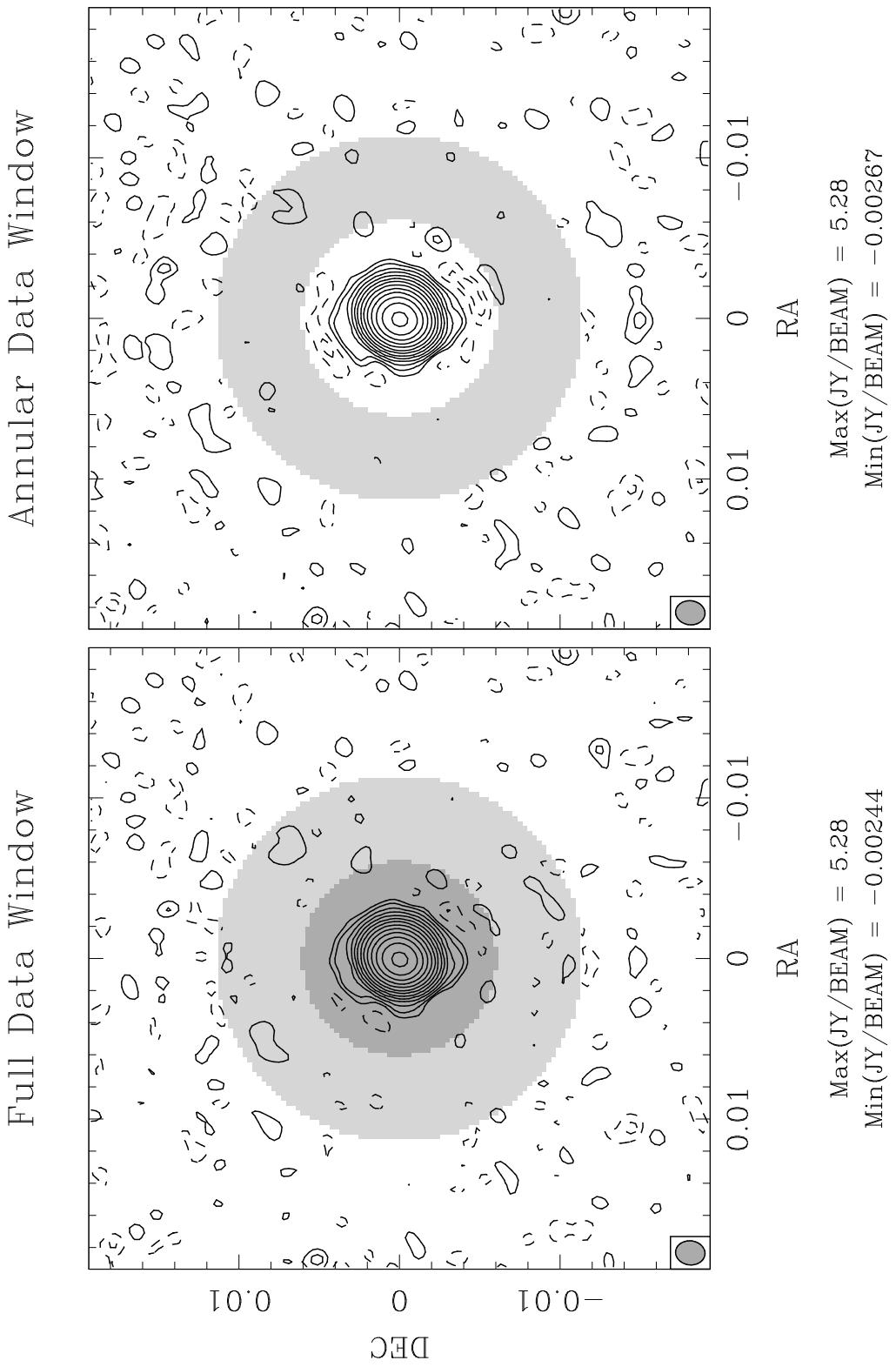


Figure 4.10: Missing data deconvolution: On the left, the algorithm has been given all pixels on the dirty map interior to the large circle as input to the deconvolution. It was allowed to place flux within the small circle. On the right, it has been given only the pixels corresponding to the shaded annulus as data, but has reconstructed the source from the sidelobes alone.

4.3.4 Discussion

NNLS has plenty of limitations and some problems. But when it works, it works *extremely* well. In an absolute sense, it works best on extremely compact emission. Compared to the other algorithms commonly used in radio astronomy, it best distinguishes itself on sources too extended for CLEAN to process well, and too compact for MEM. It has no difficulty with sharp edges, which make it a good choice for planetary images and source involving a shock. It is also very good at producing models for self-calibration. Both of these properties are presumed related to the fact that NNLS zeros the residuals in the dirty image nearly completely. Some algorithms like MEM make no particular attempt to zero the residuals completely, and simply fit the data to within the thermal noise. Unfortunately, if the residuals are correlated in the image plane, this may result in significant systematic bias at different locations in the u - v plane — very bad news for self-calibration. CLEAN will indeed zero the residuals in the asymptotic limit, but at finite iterations may also suffer a systematic bias in the u - v plane. NNLS makes a choice about the model, and then essentially jumps right to the solution. This can be a good thing or a bad thing, depending on the quality of the data, but under-iteration is rarely a problem in NNLS images. And even with quite noisy data, the solution appears robust.

For practical purposes, the NNLS algorithm lacks a termination criterion, or any other tuning parameter. This is a good thing, since any parameter which must be set for proper operation can be set incorrectly. NNLS certainly exceeds the capabilities of CLEAN and MEM on objects to which it is well suited. But also, it has the advantage that it is difficult to misuse and generally produces the best image of which it is capable. The other algorithms can produce sub-optimal images due to insufficient iteration or poorly chosen parameters. A major strength of NNLS as a practical deconvolver is that it is largely “hands off.”

On the negative side, NNLS shares the tendency of CLEAN to compact flux. A region of smooth flux will typically result in a model with regularly spaced components at intervals of a few pixels. The isolated pixels will contain all the flux that should be contained in the zero pixels surrounding them. This behavior seems to be a common failing of algorithms that operate on small numbers of pixels at a time.

The Kuhn-Tucker conditions do not guarantee a unique solution to the deconvolution problem, merely “a” solution. Clearly, since the performance of the NNLS algorithm on noiseless, discretely sampled model data is less than perfect, there must be multiple solutions present that even in the perfect case are indistinguishable by the algorithm from the real thing. Examinations of the residuals in such perfect cases show that it really has reduced them to within machine precision of zero. Unless this were so, the real solution would have a calculably better fit to the data and the algorithm would not terminate. Since the algorithm typically builds up the model a few components at a time, examining the termination criteria at each step, it will stop at the first solution that satisfies the Kuhn-Tucker conditions to its precision. In some sense, then, the NNLS algorithm is selecting something

like a minimum-number-of-pixels solution. The addition of positivity and support has reduced the dimensionality of the null space somewhat, but the problem is still singular. The final implicit criterion in the algorithm is that it stops at the first acceptable solution, which is built up a few pixels at a time. It means that if the real solution has a great many smooth pixels, the algorithm will always select a flux compacted solution over the true answer, so long as both are indistinguishable from the point of view of fitting the data. It definitely would be worth looking into other algorithms that solve the same in different ways, to see if variations in how the solution is computed make a significant difference.

The NNLS algorithm is used in the simulation Chapter 5, the 3C48 case study Chapter 7, the SN1987A case study Chapter 8 and the DA193 case study Chapter 9, so we will see a good deal more of its properties by example. The SVD deconvolution is only used henceforth in Chapter 8.

Chapter 5

Simulations

This chapter is composed primarily of a number of deconvolution simulations. The thrust of all of these simulations is to give a lower limit error bound. All of these simulations are noiseless, and except where representation issues are fundamental to the simulation, all have been discretized in the image plane. That is, the model of the source was sampled onto a regular pixel grid, the point spread function treated likewise, and both were discretely convolved to create the dirty image used as input to the deconvolution algorithm. The effect of this is that the problems set before the deconvolution algorithms possess an answer that obeys the assumptions of bounded support and positivity, unlike real sources which may violate these constraints at a low level due to discrete representation effects. Thus the imaging performance on real sources may well be worse than given in these simulations, for a wide variety of causes including thermal noise, calibration error, and representation effects. The imaging performance is highly unlikely to be *better* than these simulations, however, and the results given here show deconvolution errors occurring at a significantly higher level than many people believe. Whether recognized or not, errors at the levels given here or higher are probably present in any astronomical image deconvolved with one of the algorithms listed below.

The bulk of the chapter is a number of simulation series all designed to address the issue of deconvolution behavior on simple isolated sources. These are the “resolution tests,” since most of these simulation series are indexed by at least the size of the model relative to the point spread function. This meta-series begins with a catalog of error magnitudes and morphologies produced by deconvolving an isolated Gaussian or Disk model source using a Full Track VLA point spread function, taken from an actual high quality observation. For each of the three major algorithms used in this dissertation, CLEAN, MEM and NNLS, and each of three model sizes, the magnitude of the model error in the u - v plane is given in both greyscale and quantitative slice format. Additionally, the component model for each case is displayed as an image, as is the error in the sky plane between the smoothed model and reconstructed image. A similar series of deconvolution tests is given with the same model sources, this time varying the source smoothly through two orders of magnitude in size, and exploring a wider parameter space of point spread functions and algorithms. The result of each is graphically presented by the error quantities in the nominal-resolution reconstructed image. The residuals in both planes are presented, along with the quantities “maximum error in the reconstructed image”, “maximum error off the support of the source”, and “RMS

error off the support of the source”. There is a series against differing point spread functions for the major algorithms, and a cross comparison series against different algorithms returning to the Full Track VLA PSF and adding in the algorithms Maximum Emptiness, SDI CLEAN and GSP. The width of the final restoring beam was found to be extremely important in the reconstruction fidelity, and the error quantities against algorithm at a particular source size are presented varying the deconvolution width from half the nominal to twice the nominal width. Finally, a scaled version of the model 3C48 source is used with CLEAN & MEM only, presenting maximum error only, as a demonstration that a somewhat more realistic model source yields results similar to those of the isolated single component models.

The thrust of the resolution test meta-series changes slightly as we next present not error quantities in the restored image, but error quantities in the restored summed flux over the support of the source. This is again indexed by model size and all six algorithms, and is presented for the Full Track VLA PSF and Snapshot VLA PSF, for both the Gaussian and Disk models. Sadly, this series only addresses a nominal deconvolution width. Last of all in the meta-series, we consider a modification to CLEAN, where the sky is considered composed of Gaussians rather than delta functions. This uses the Full Track VLA PSF again, and indexes against model size, model type, pixel size and convolution width. In an uncharacteristic nod to brevity, only the maximum error and RMS plots are presented.

Numerous as are the simulations and quantities presented, these plots are really only slices through a larger parameter space explored. The relatively small ASCII files containing the numerical data values which generated the plots in the resolution test meta-series will be made available for the indefinite future as part of the SDE software distribution. See Appendix H for a description of the “Electronic Supplement” and how to obtain it.

The chapter concludes with two final series which simulate their data in the u - v plane. Both use a synthetic Full Track VLA PSF. The first addresses the question of a shifted point source, which happens to lie halfway between two pixels in the sky plane discretization. It is shown how well a perfect deconvolution algorithm could reconstruct such a source for two different oversampling factors, and then how well the three major algorithms do on noiseless u - v data. The last simulation series in the chapter addresses the issue of how quickly CLEAN and MEM converge, as a function of position in the u - v plane. Both are run on a shifted point source, for two different oversampling factors, and with and without a Gaussian pixel function.

5.1 The Resolution Test Deconvolutions

This section describes most details of the deconvolutions performed in the resolution test meta-series. A few later variations are described in the running text. The algorithms themselves are described in Section 2.7.

Most test series use the point spread function from the observation of 3C48 given in Figure 3.14, with observation parameters and u - v coverage in Table 3.7 and Figure 3.12a, respectively. It is a realistic example of a good single-configuration VLA observation of a high declination source. The fitted beamwidth is $.208'' \times .181''$, for a mean beamwidth of $.194''$. The dirty images were of size 512^2 and pixel spacing $.03''$. Model sources were constructed with full-widths at half-maxima of $3^{n/4}b$, where b is the basic width of the series. In this case $b = .196''$. In the full series, n varied by integers from -9 to 7 for most algorithms and from -9 to 4 for NNLS. The extreme limits of smoothness were considered by using a Gaussian and Disk as the model, which probably have edges smoother and sharper respectively than any source likely to be encountered in practice. The models were azimuthally symmetric. It should be mentioned that MEM has a somewhat unfair advantage when run on the Gaussian models, in that the $-I \ln I$ form of the entropy can be shown analytically to produce Gaussian peaks around sharp features in the image, (Narayan and Nityananda, 1986). But CLEAN possibly has a similar advantage in the Disk models, in that it is not burdened with an inappropriate smoothness criterion. About all that can really be said is that the behavior on true sources will likely lie between these extremes.

The CLEAN deconvolutions for the cases $n \leq 3$ were run to 40,000 components at a loop gain of 0.1. The $n = 4$ and 5 cases were run to 100,000 components, $n = 6$ to 150,000 components, and $n = 7$ to 200,000. The algorithm was allowed to place components wherever it chose, and this was verified to make no significant difference in the solution quality. In general, the support constraints are more important for noisy and ill-calibrated data, but in these tests it was not usually an important consideration. While the number of iterations was rather arbitrary, these choices were sufficient to reduce the image plane residuals to a small fraction of the RMS reconstruction error. Typically the maximum residual was down by 10^{-5} from the peak smoothed model. The nature of the residuals varies considerably with algorithm. CLEAN and NNLS reduce the residuals to near zero, which may or may not be an advantage depending on the nature of the problem and data. The other algorithms generally do not. This is discussed in more detail in Section 5.3 where a plot is presented of RMS image plane residuals.

All MEM deconvolutions in this series were run to 100 iterations each, which is comfortably in excess of the canonical 40 iterations typically used. This was sufficient that the $n > 4$ cases had converged completely, and the $n = 2, 3, 4$ cases nearly so. For emission more compact than that, the algorithm had entered a regime where tiny improvements in entropy and fit to the data are traded off against each other. There is a *very* gradual improvement in the solution quality at this point. For practical purposes, the algorithm may be said to have converged. By spot checking individual cases, the solutions improve by a small factor when run to extreme iteration, but not by enough to affect any qualitative description given here. Since there is no thermal noise in these simulations, an arbitrary level had to be selected for the agreement parameter between model and data. These simulations used $\text{Sigma} = 10^{-6}$, and allowed the algorithm to estimate the total flux by

itself with `Tflux = -.1`. Again, spot checking individual cases showed only the weakest dependence on the exact value of the parameters selected, and these results are expected to be generic for noiseless compact emission.

No termination or tuning parameters were needed for the NNLS deconvolutions. Though in some cases the number of iterations exceeded the $3N_{flux}$ limit proposed by Lawson and Hanson, the algorithm was run until the solution satisfied its stopping criterion. NNLS does require support information to make the problem computationally tractable, however, and was allowed to place flux in the region where the model, smoothed to the fitted PSF, exceeded 10^{-5} of its maximum. The NNLS data window was a circular region centered on the source, 40 pixels in radius for a total of 5013 pixels, or the flux window, whichever was larger.

Many of the more detailed results examine only the algorithms CLEAN, MEM and NNLS. The graphical error plots also examine the Maximum Emptiness (L1) algorithm, SDI CLEAN and Gerchberg-Saxton-Papoulis. The L1 deconvolutions were very similar to the MEM deconvolution with the parameters of 100 iterations, `Sigma = 10-6`, and `Tflux = -.1`. The SDI CLEAN deconvolutions are also measured in “components,” the only thing changing being how the components are selected. The same number of components were used for SDI CLEAN as with CLEAN. The `Trim` parameter was 0.5, and the loop gain again 0.1. Finally, the GSP algorithm was run to 1000 iterations for all cases, with an acceleration parameter `Accel` of 0. Like NNLS, the GSP algorithm essentially requires support information. (In many cases, the deconvolution diverged without it.) The same support window of 10^{-5} of the peak smoothed model was used. No support window was used for the other four algorithms.

5.2 Error Morphology

Figures 5.1 through 5.7 attempt to characterize the morphology of the errors made by the three major algorithms on the model sources. Here we present detailed images of the $n = -2, 2$, and 6 cases, for a beamwidth-to-model-width ratio of $.58, 1.73$ and 5.20 respectively. The $n = 6$ case exceeded the limits of the NNLS algorithm, and the $n = 4$ case is used instead, with a ratio of 3.00 . The most dramatic of the plots provided for each case is a grey scale plot of the error pattern in the Fourier plane. This is simply $|\mathfrak{F}[M - C]|$, where M is the model and C the component representations produced by the algorithms. Notice that the residuals are *not* included in these plots. Each panel of each plot has been allowed a full independent linear range of greyscale, to better show the morphology of the error pattern. The amplitude of the error pattern is symmetric due to the Hermetian nature of both sky and component model, and only the Fourier half plane is shown out to a $\hat{u}_{max} = \hat{v}_{max} = 3.43 M\lambda$. The phase information is also interesting, but not sufficiently so to include here. Color GIFs of these plots showing both amplitude and phase can be found in the electronic supplement to this dissertation, as documented in Appendix H.

The three lobe figure traced in white in each panel of the Model Error Amplitude plots is the envelope of the VLA sampling pattern which generated the beam, and the three straight lines in each are given quantitatively as slices in the lines plots on the facing page. In these line plots, the solid lines are slices through the error amplitude plots as indicated, with heavy dots indicating the point where the slice crosses the sampling envelope. The large dashed line is the transform of the model, and the upper light dotted line is the transform of a typical restoring beam, $.2''$ in this case. The lower light dotted lines are the same slices after smoothing down by the restoring beam, again with a dot where the slices cross the sampling pattern. Each case also includes grey scale plots of the model components and the error pattern in the restored image at a resolution of $.2''$. Both the component model and error pattern are displayed on a logarithmic grey, either zero to the maximum or two sided logarithmic around zero, depending on whether or not the particular image can assume negative values. The dotted circles in these plots are locations where the model smoothed to $.2''$ reaches $10^{-1}, 10^{-3}$, and 10^{-5} of the peak smoothed value, so the outer circle also corresponds to the NNLS support window. One final figure at the end of this set is a cross-comparison plot for the different algorithms, presenting model error slices for the Gaussian $n = 2$ case on a linear scale instead of a log scale. This plots shows the sharpness of the error pattern at the sampling envelope more dramatically than the quantitative semi-logarithmic plots.

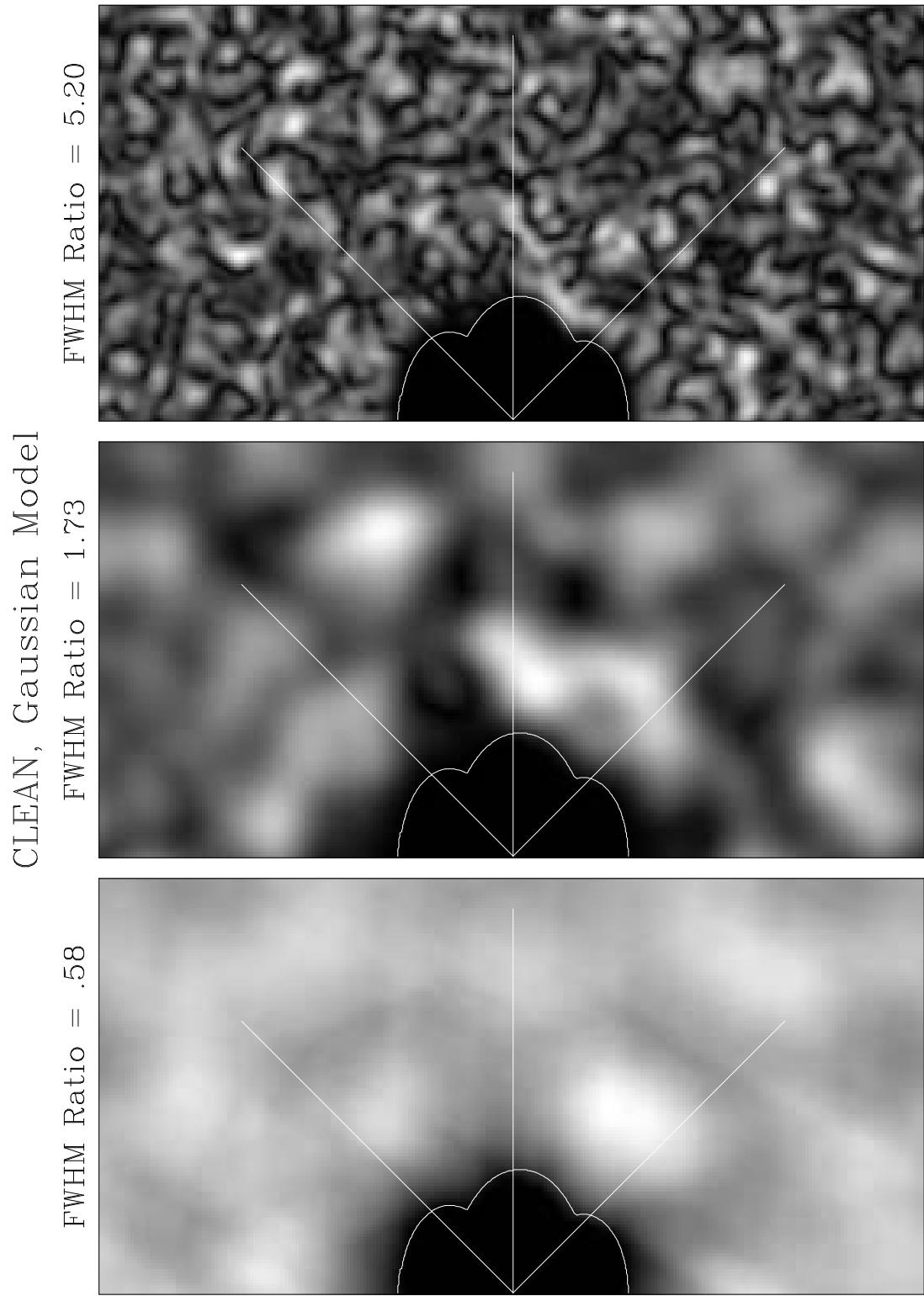


Figure 5.1a: Error Amplitude in the $u-v$ Plane — CLEAN, Gaussian Model

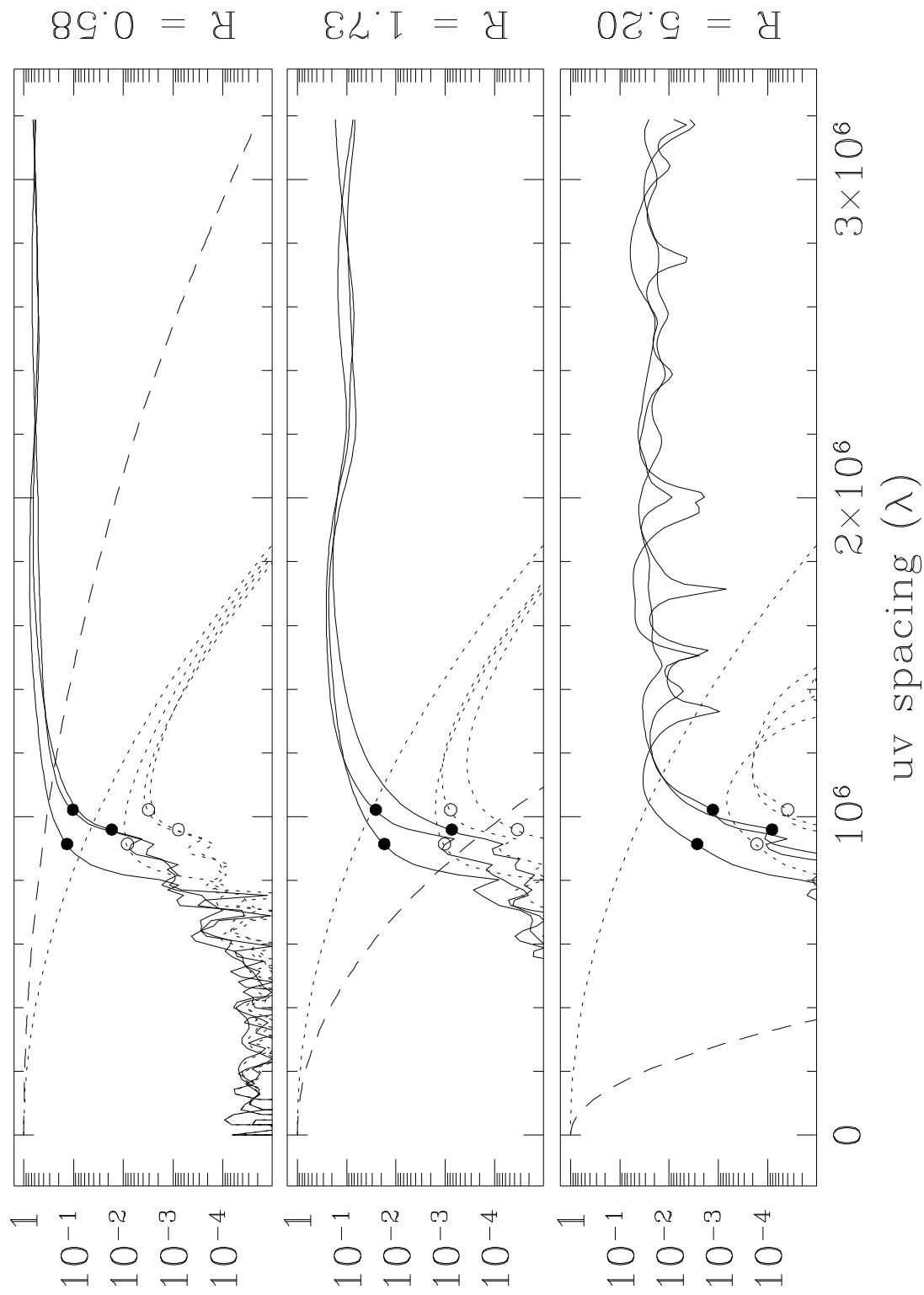


Figure 5.1b: Model Error Amplitude in the u - v Plane — CLEAN, Gaussian Model

CLEAN, Gaussian Model

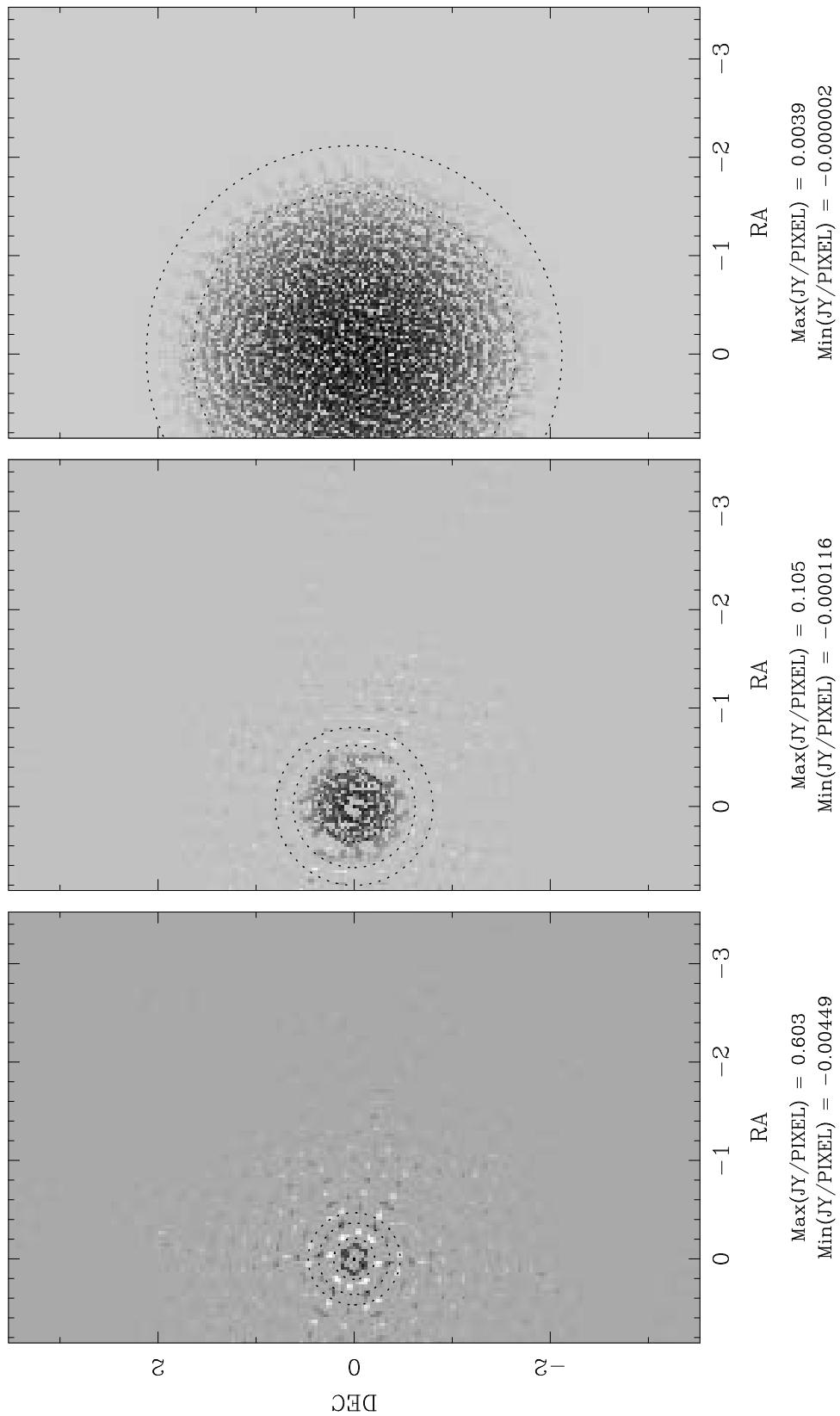


Figure 5.1c: Component Model in the Sky Plane — CLEAN, Gaussian Model

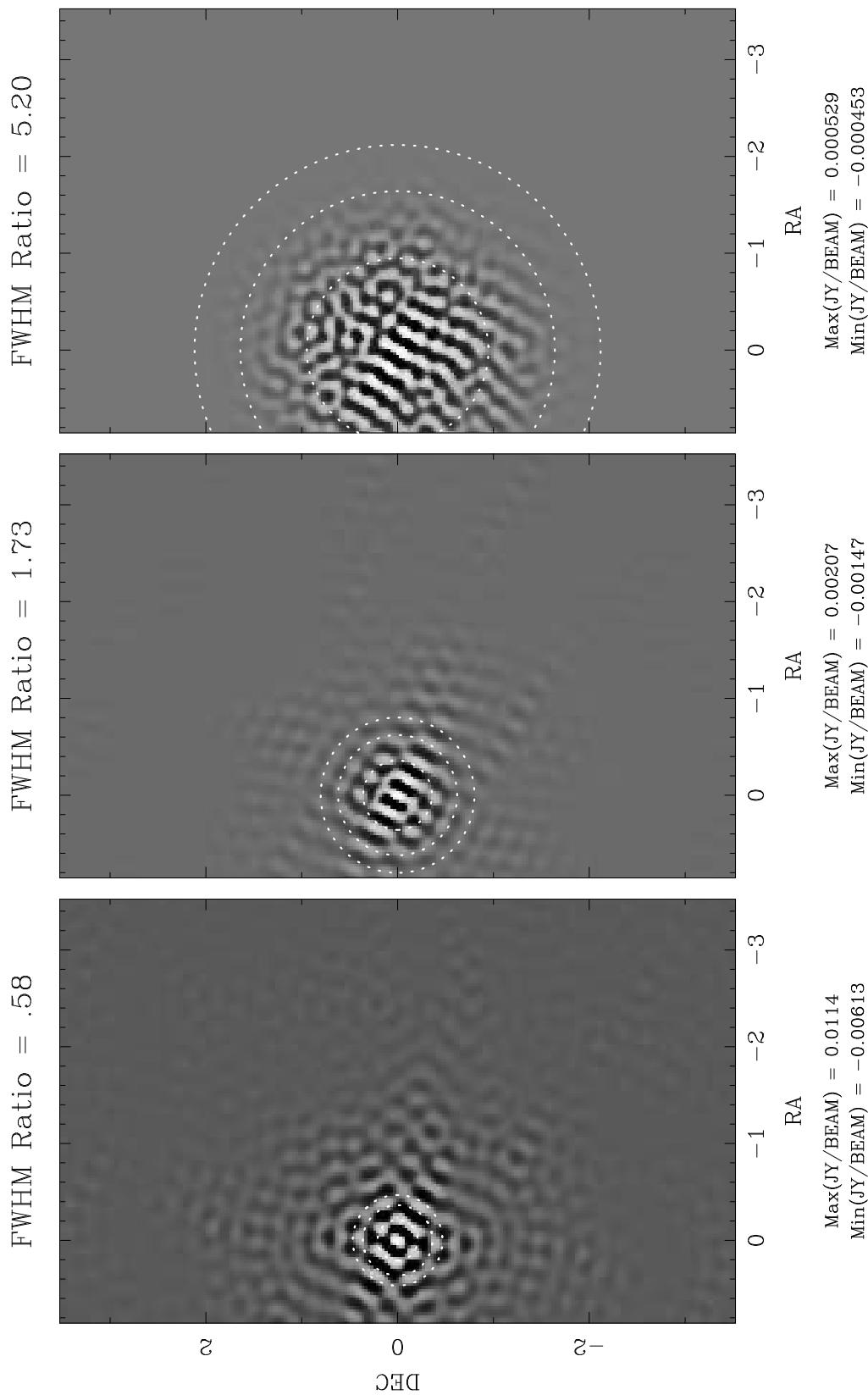


Figure 5.1d: Error Pattern in the Sky Plane — CLEAN, Gaussian Model

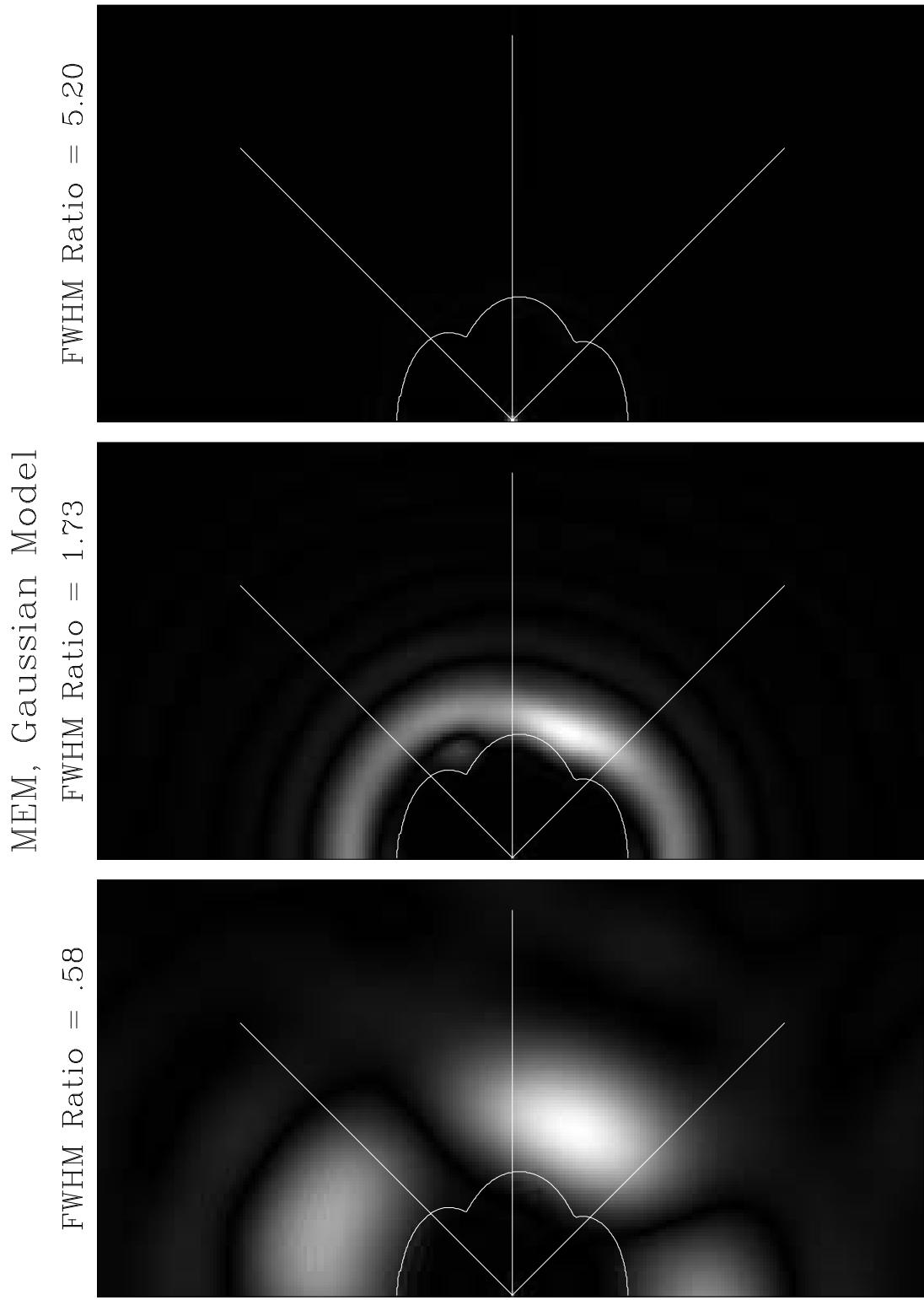


Figure 5.2a: Error Amplitude in the $u-v$ Plane — MEM, Gaussian Model

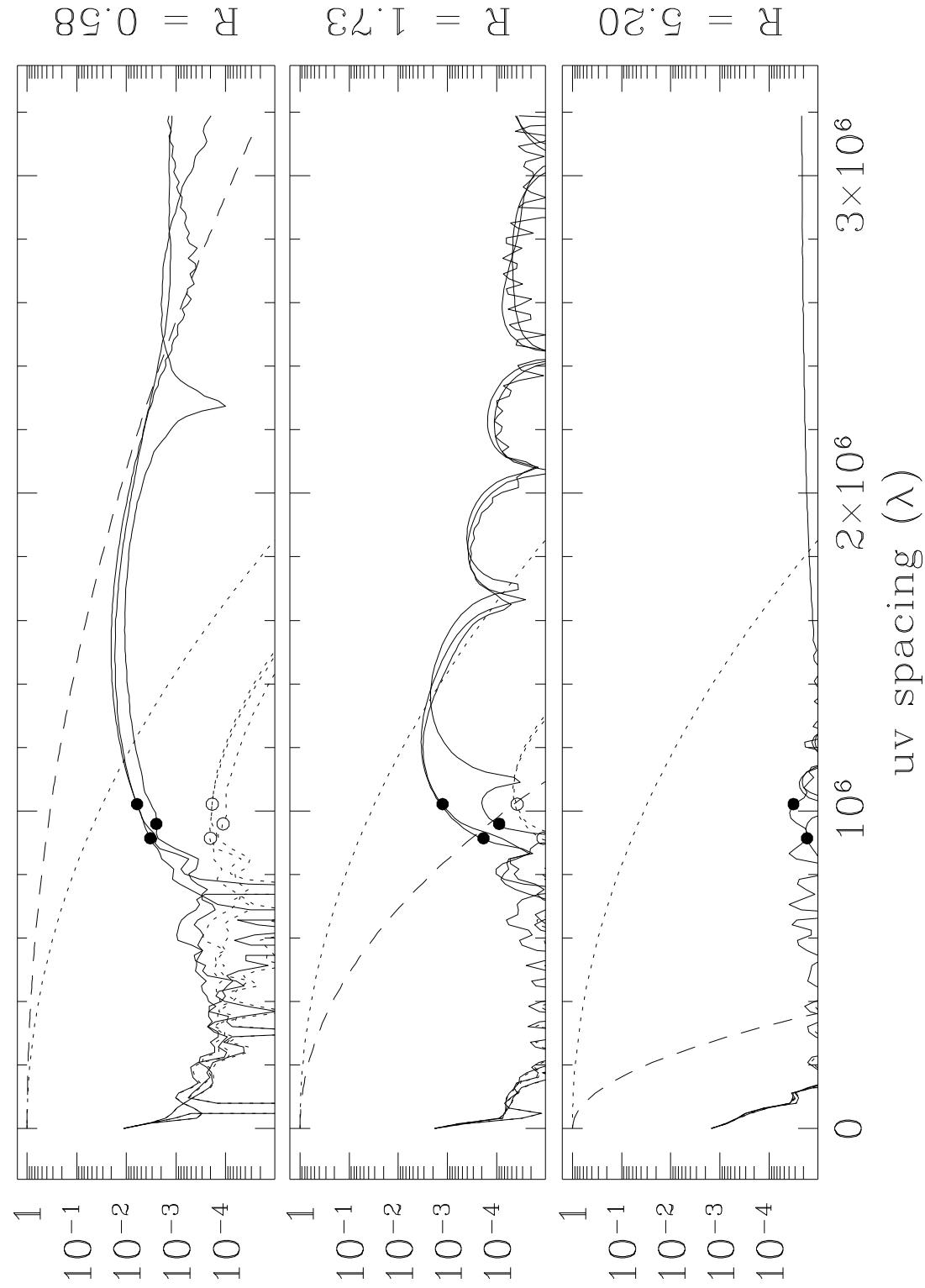


Figure 5.2b: Model Error Amplitude in the u - v Plane — MEM, Gaussian Model

MEM, Gaussian Model

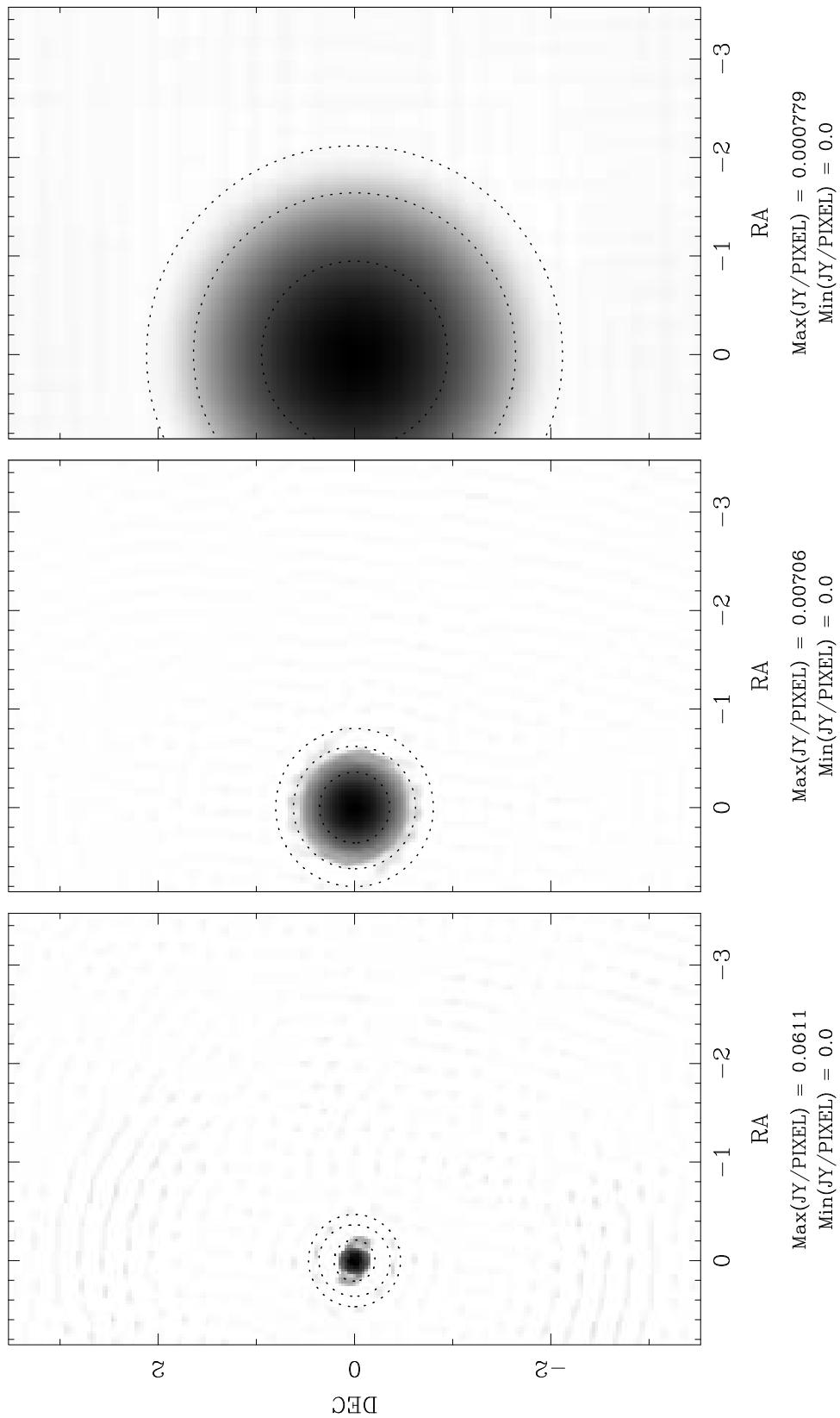


Figure 5.2c: Component Model in the Sky Plane — MEM, Gaussian Model

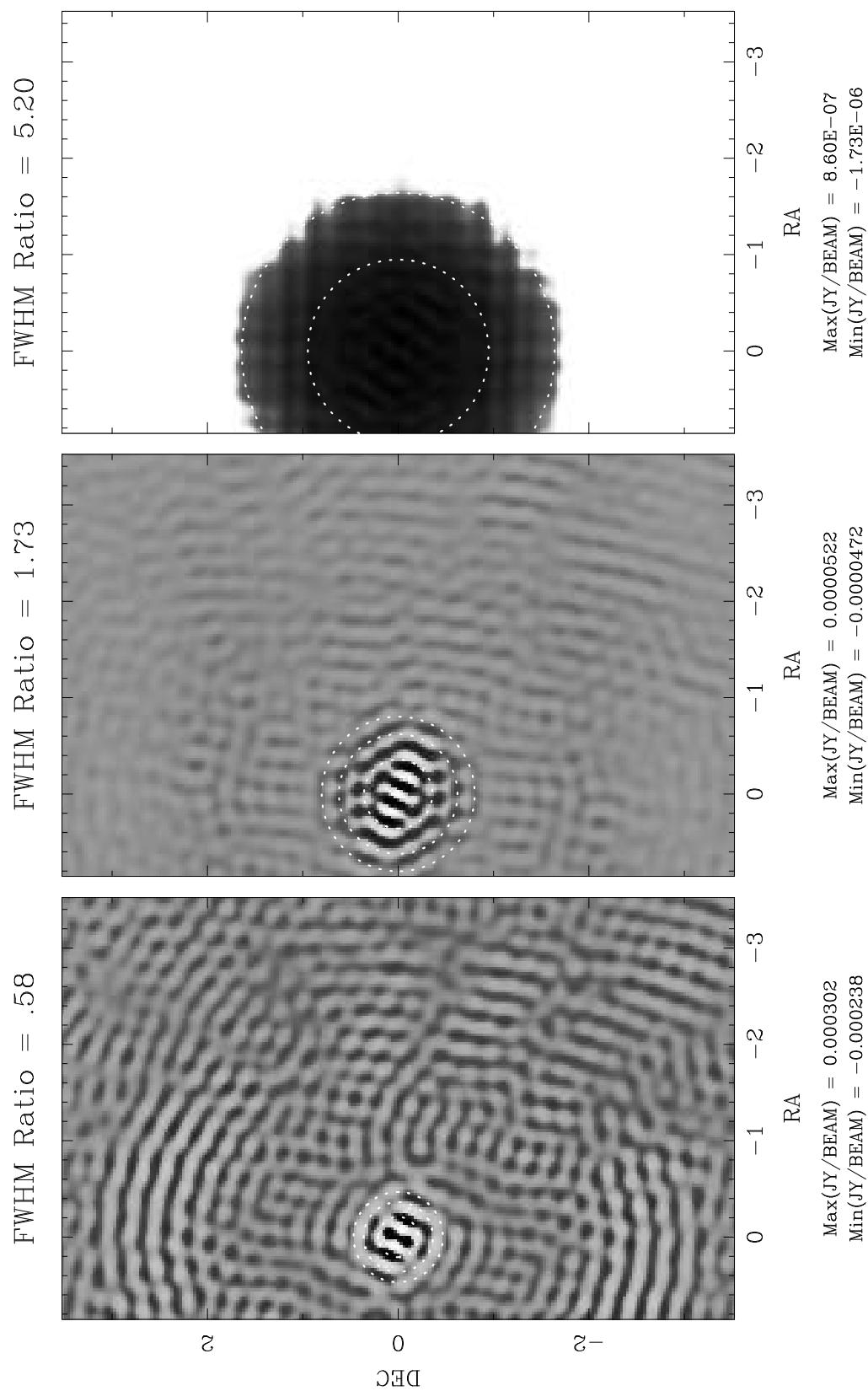


Figure 5.2d: Error Pattern in the Sky Plane — MEM, Gaussian Model

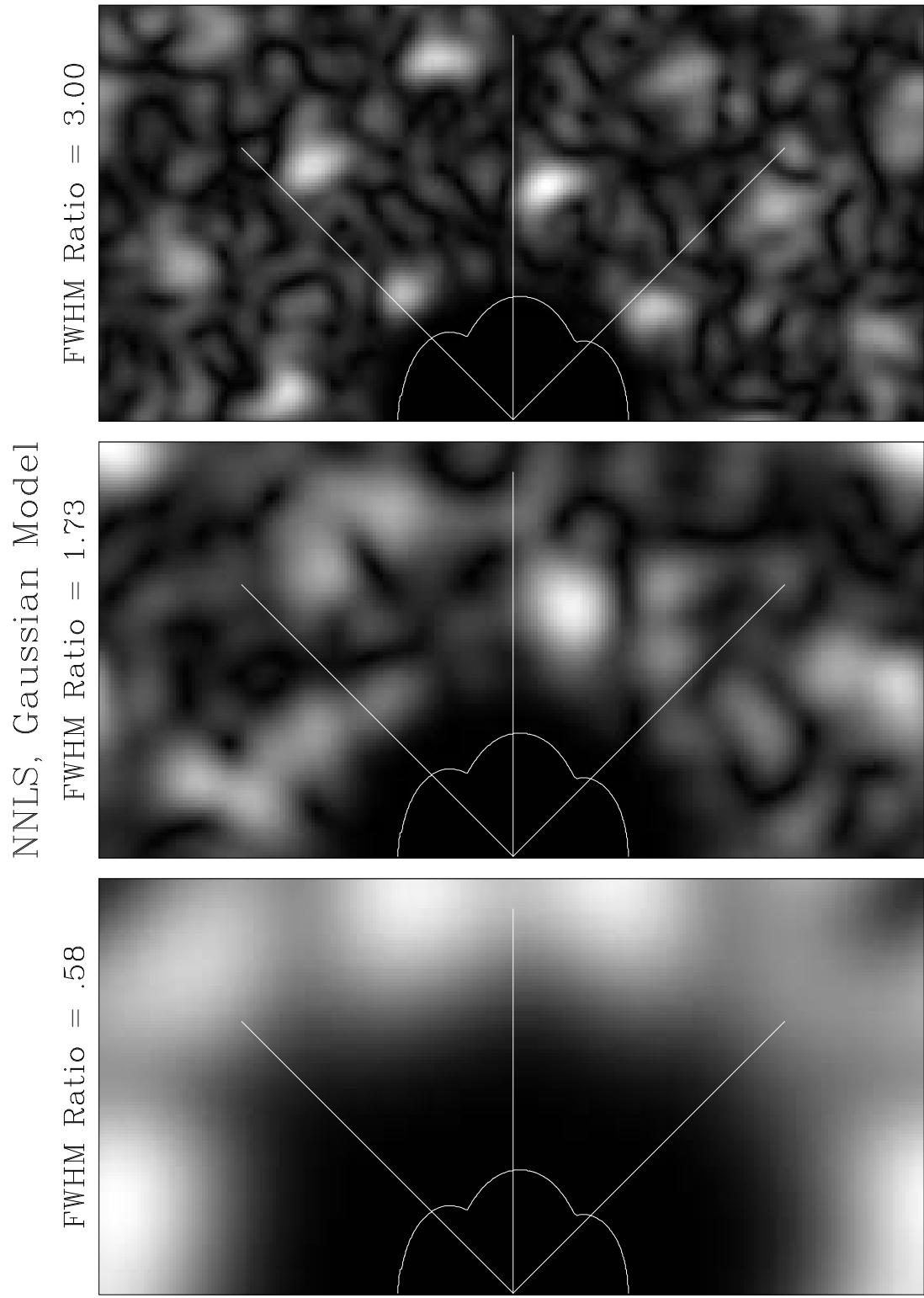


Figure 5.3a: Error Amplitude in the $u-v$ Plane — NNLS, Gaussian Model

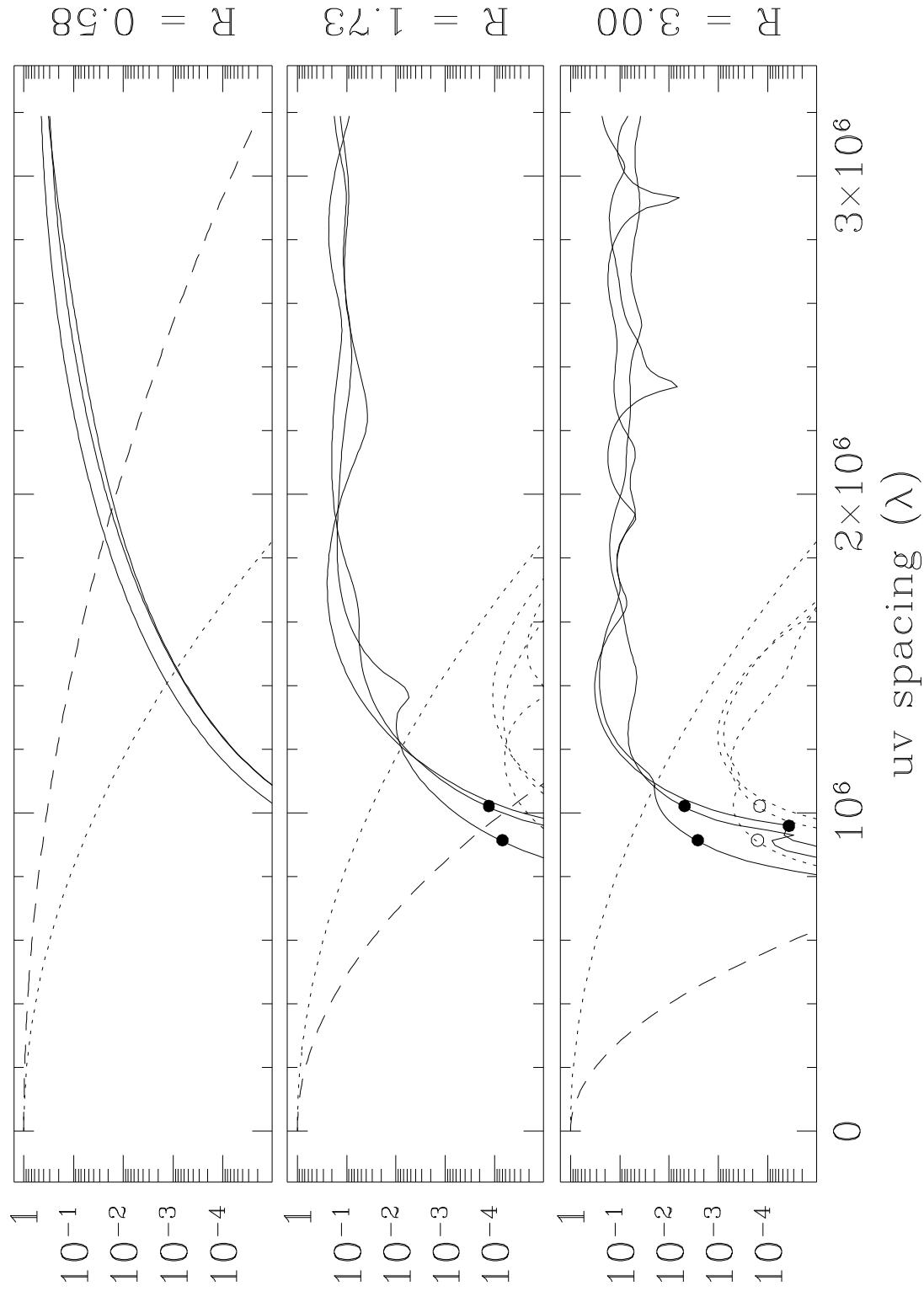


Figure 5.3b: Model Error Amplitude in the u - v Plane — NNL_S, Gaussian Model

NNLS, Gaussian Model

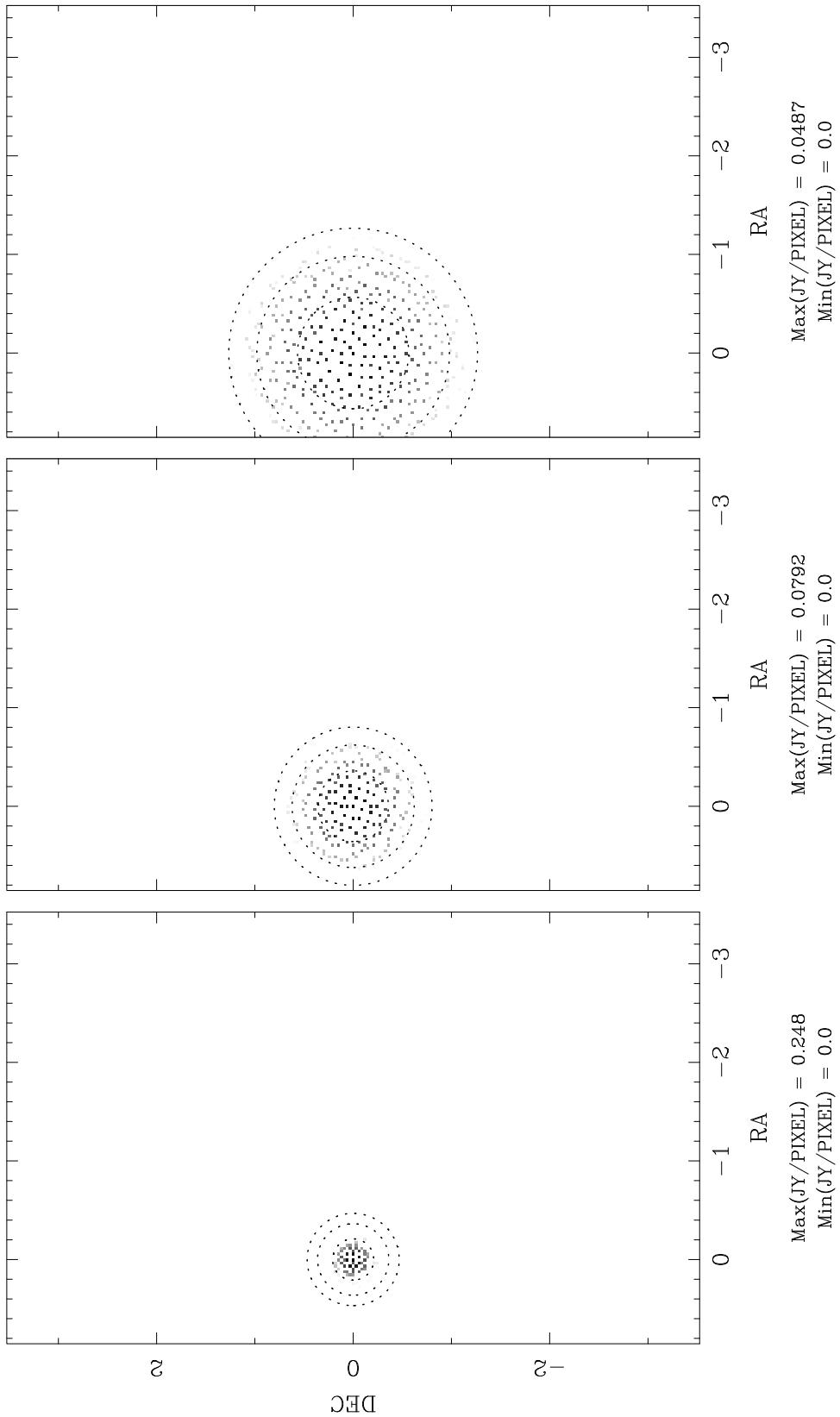


Figure 5.3c: Component Model in the Sky Plane — NNLS, Gaussian Model

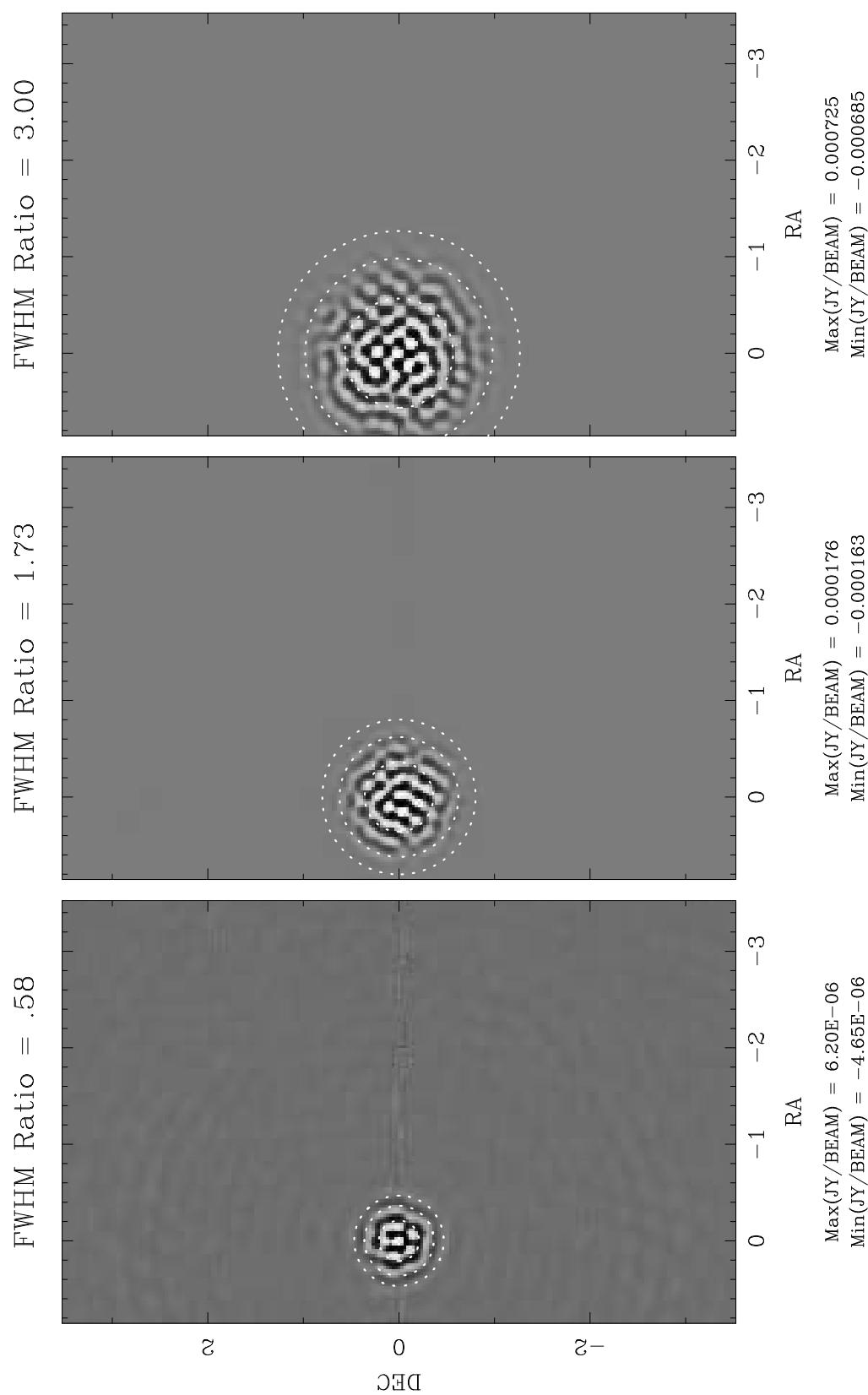


Figure 5.3d: Error Pattern in the Sky Plane — NNLS, Gaussian Model

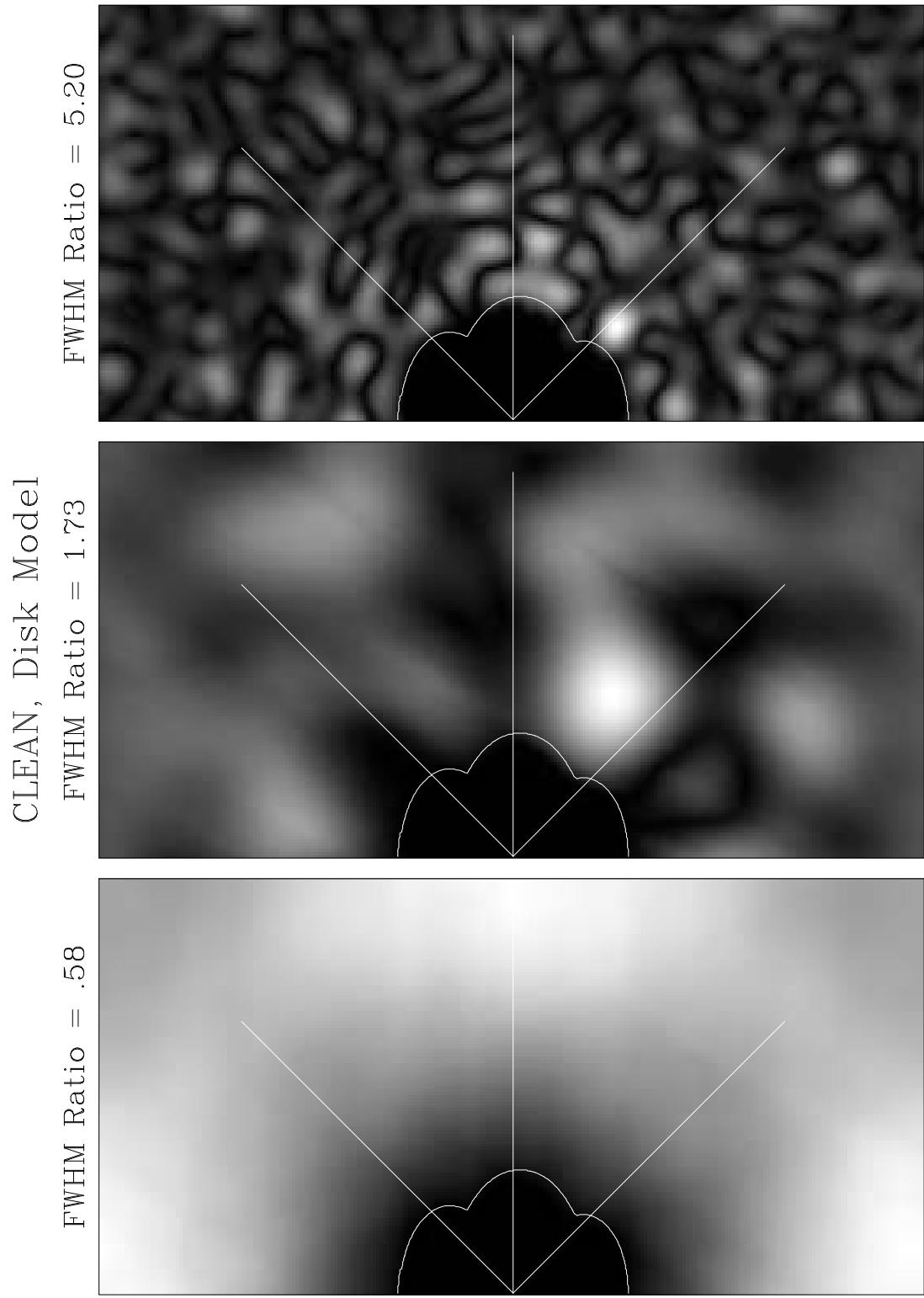


Figure 5.4a: Error Amplitude in the u - v Plane — CLEAN, Disk Model

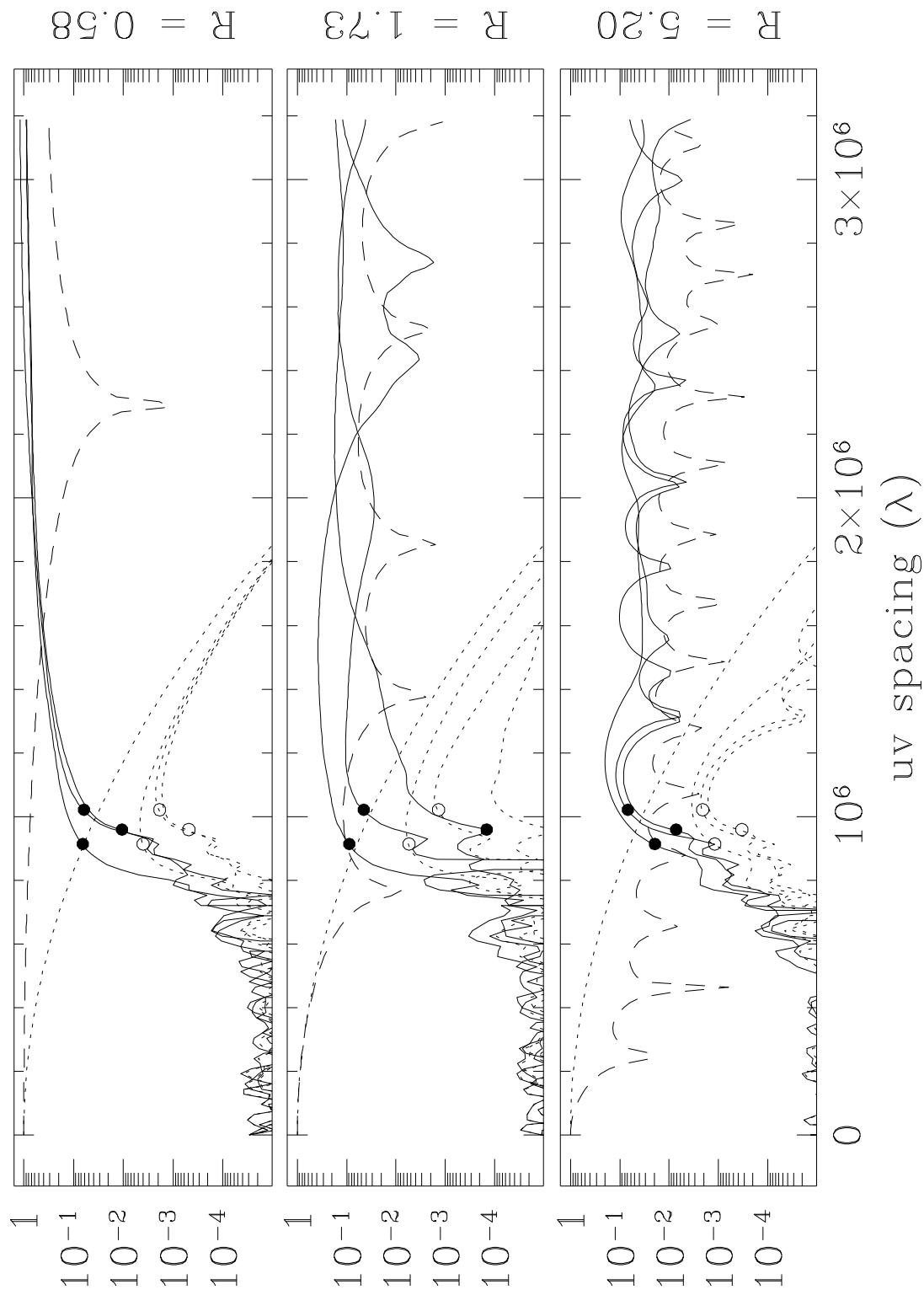


Figure 5.4b: Model Error Amplitude in the u - v Plane — CLEAN, Disk Model

CLEAN, Disk Model

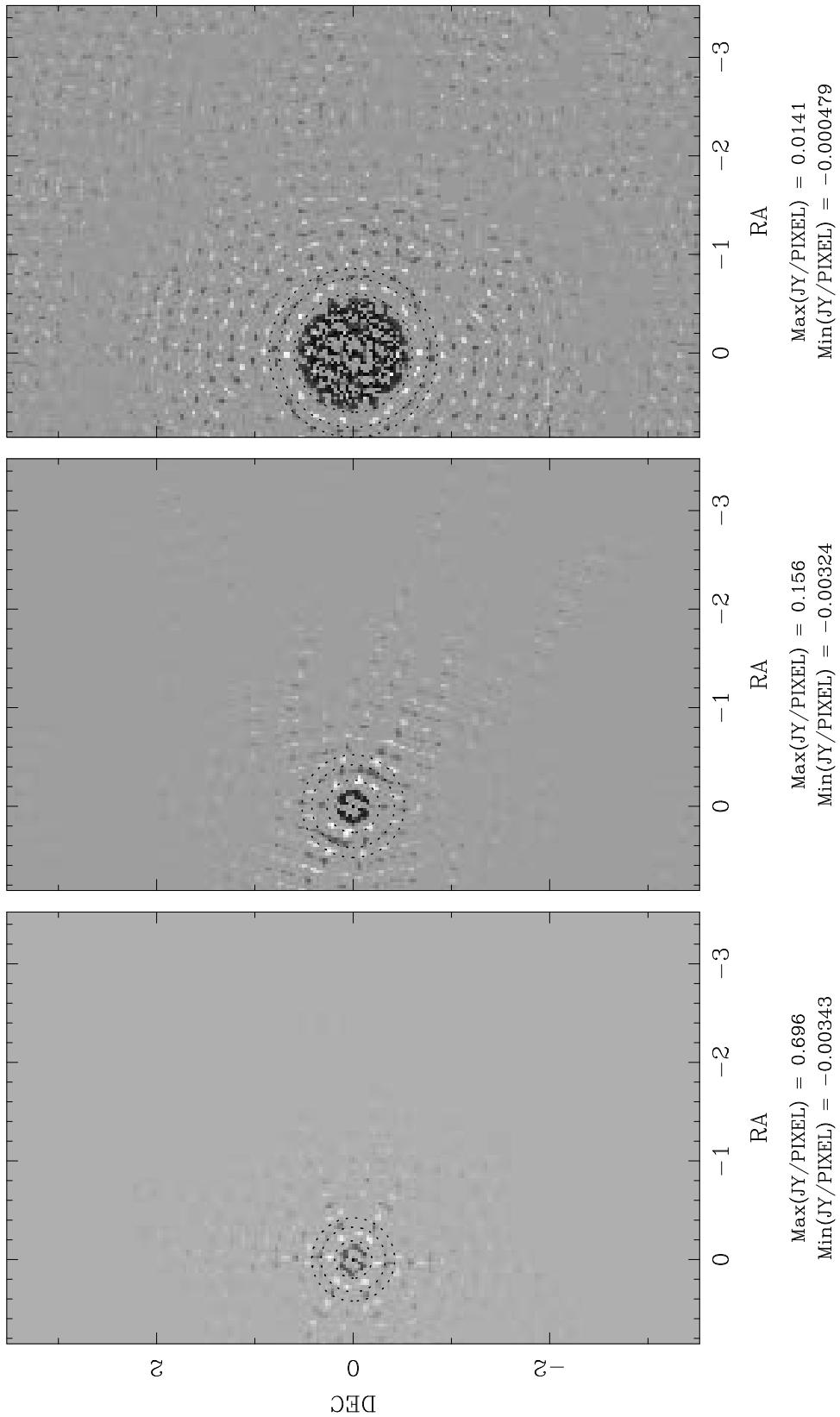


Figure 5.4c: Component Model in the Sky Plane — CLEAN, Disk Model

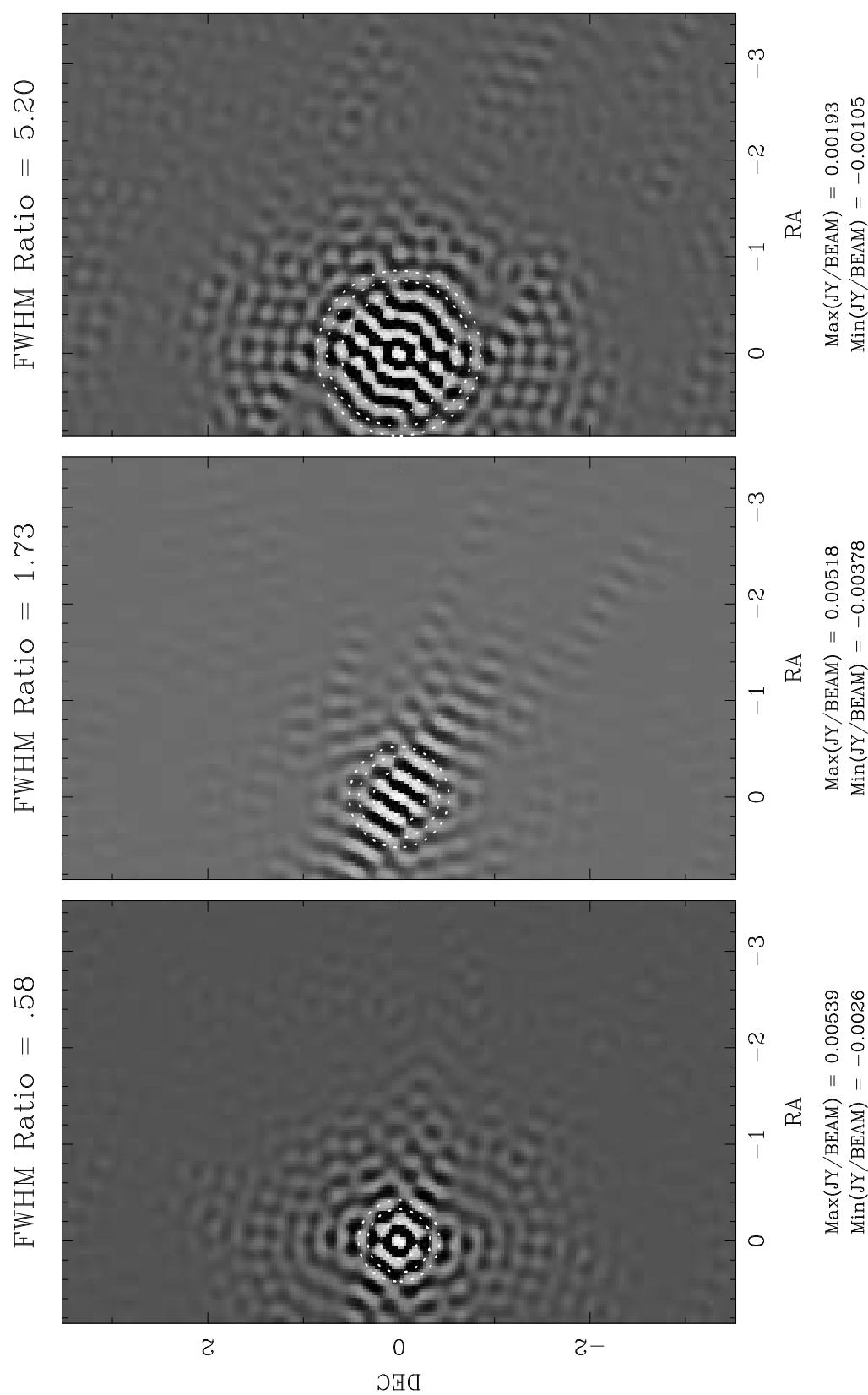


Figure 5.4d: Error Pattern in the Sky Plane — CLEAN, Disk Model

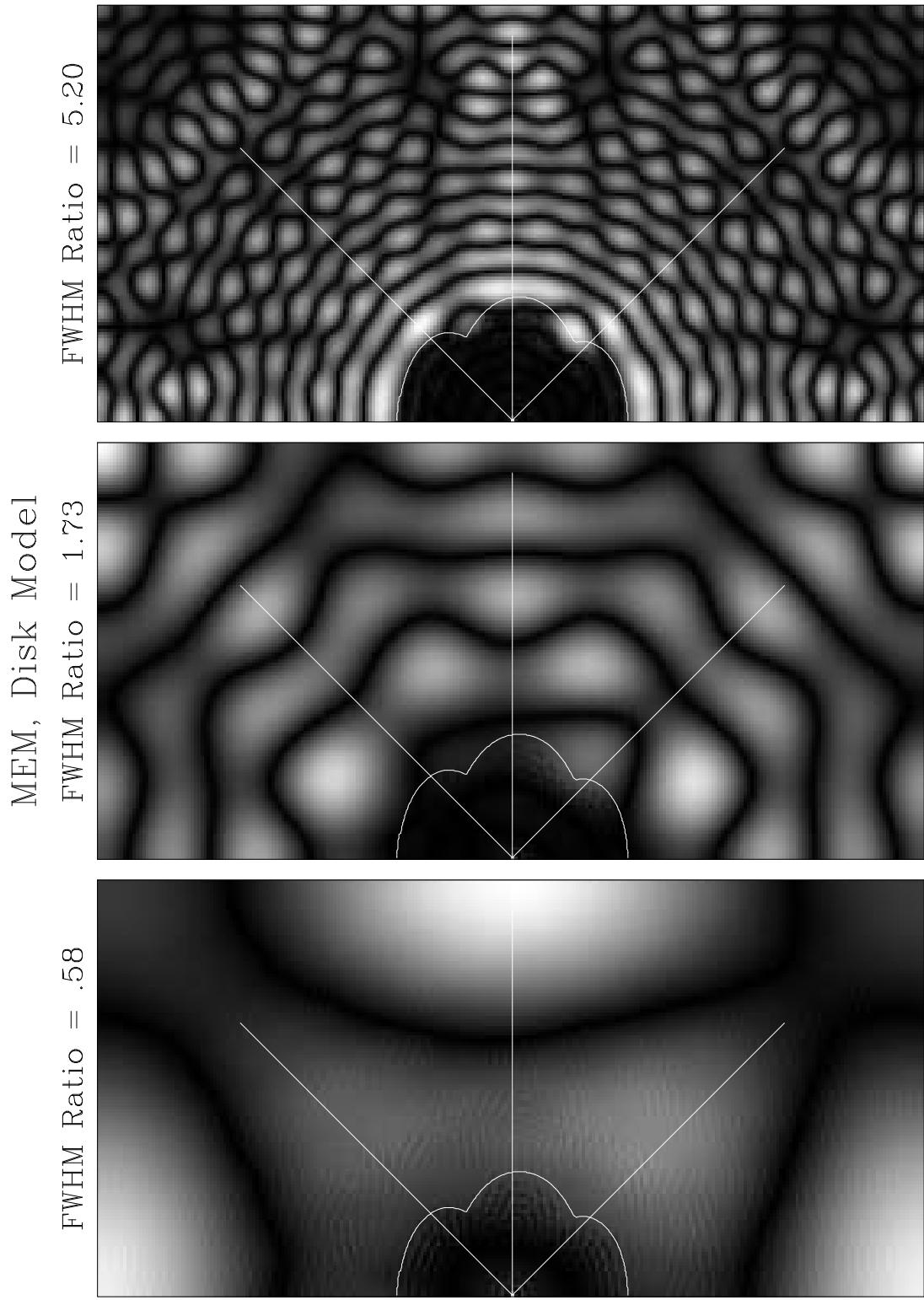


Figure 5.5a: Error Amplitude in the u - v Plane — MEM, Disk Model

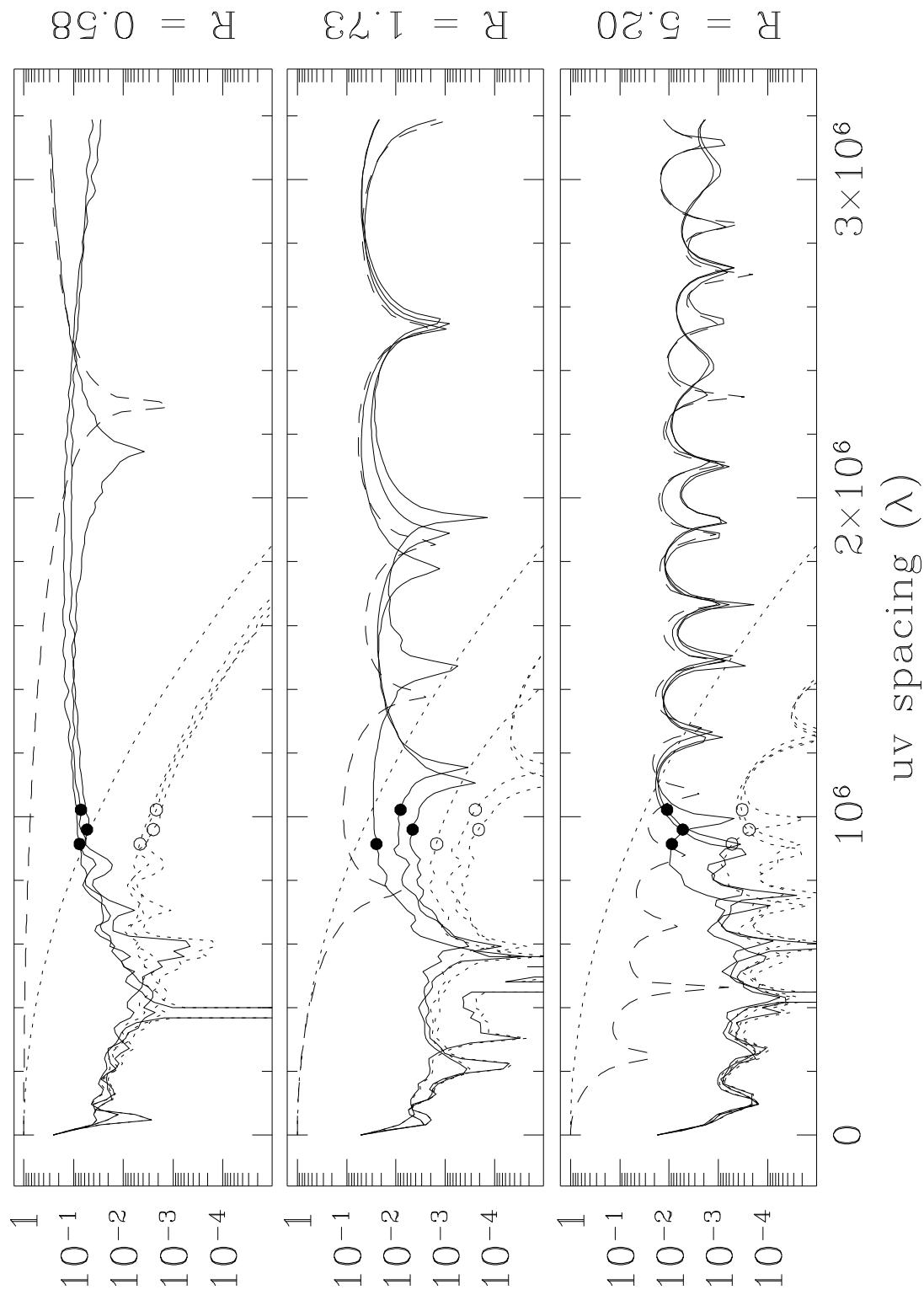


Figure 5.5b: Model Error Amplitude in the u - v Plane — MEM, Disk Model

MEM, Disk Model

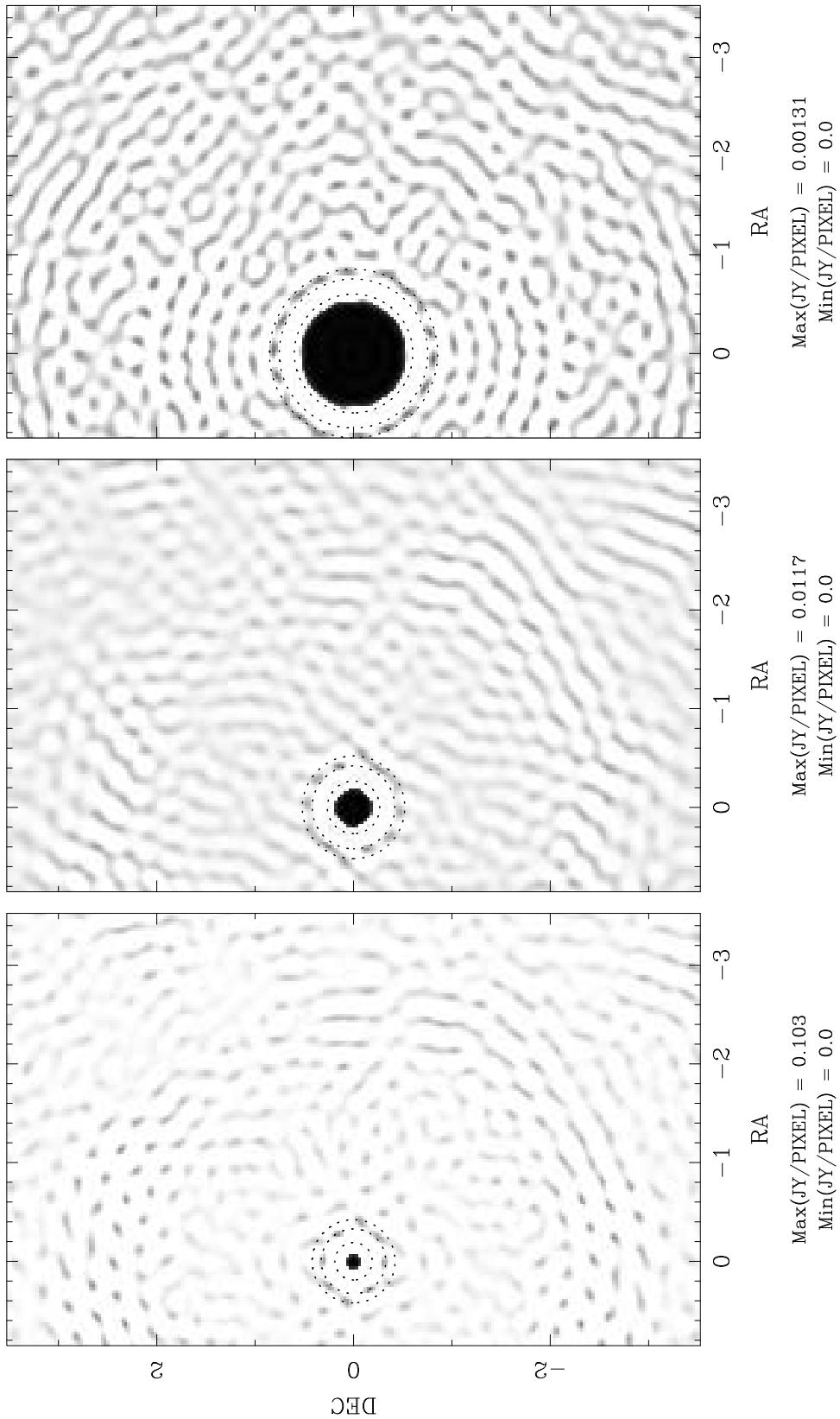


Figure 5.5c: Component Model in the Sky Plane — MEM, Disk Model

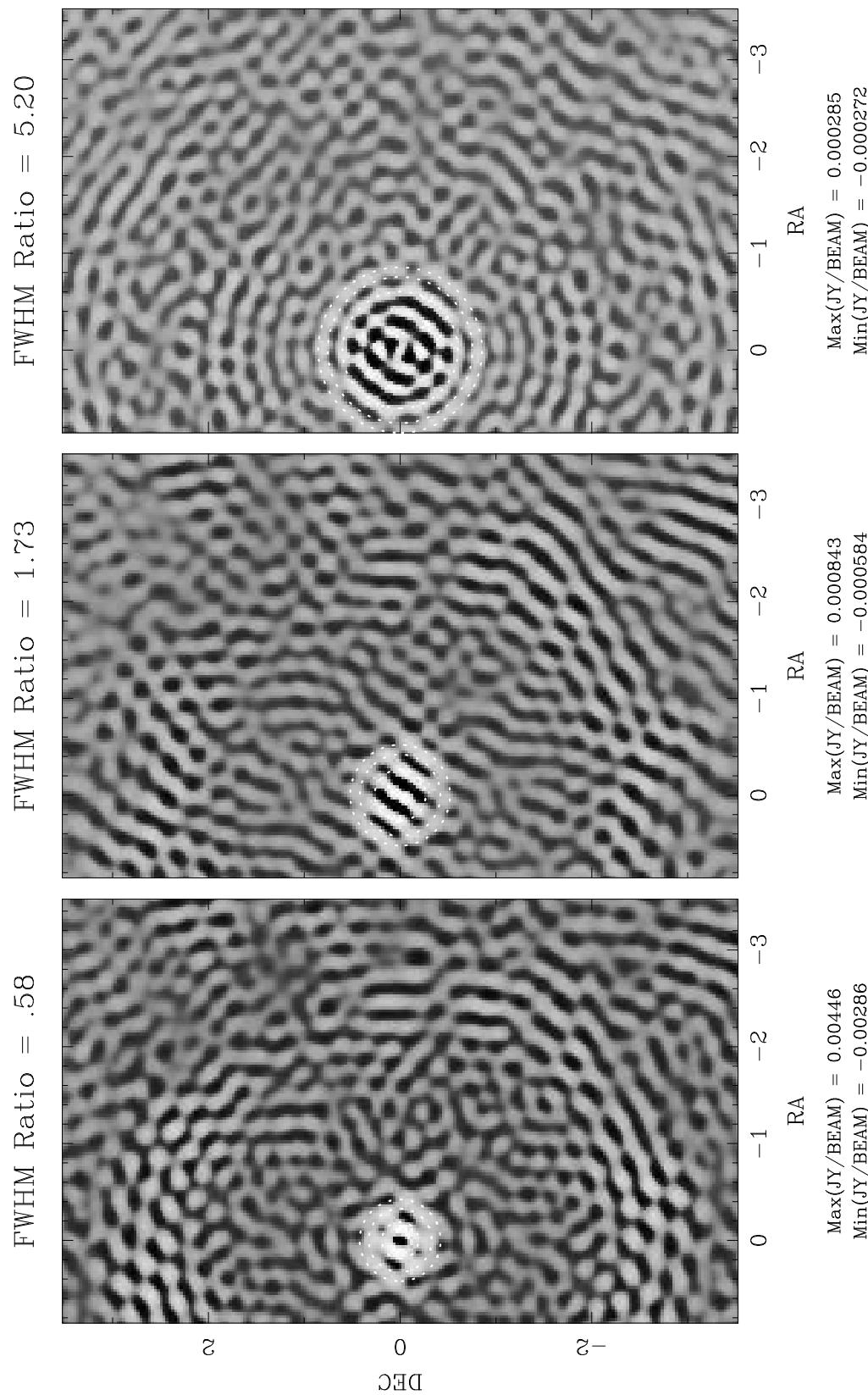


Figure 5.5d: Error Pattern in the Sky Plane — MEM, Disk Model

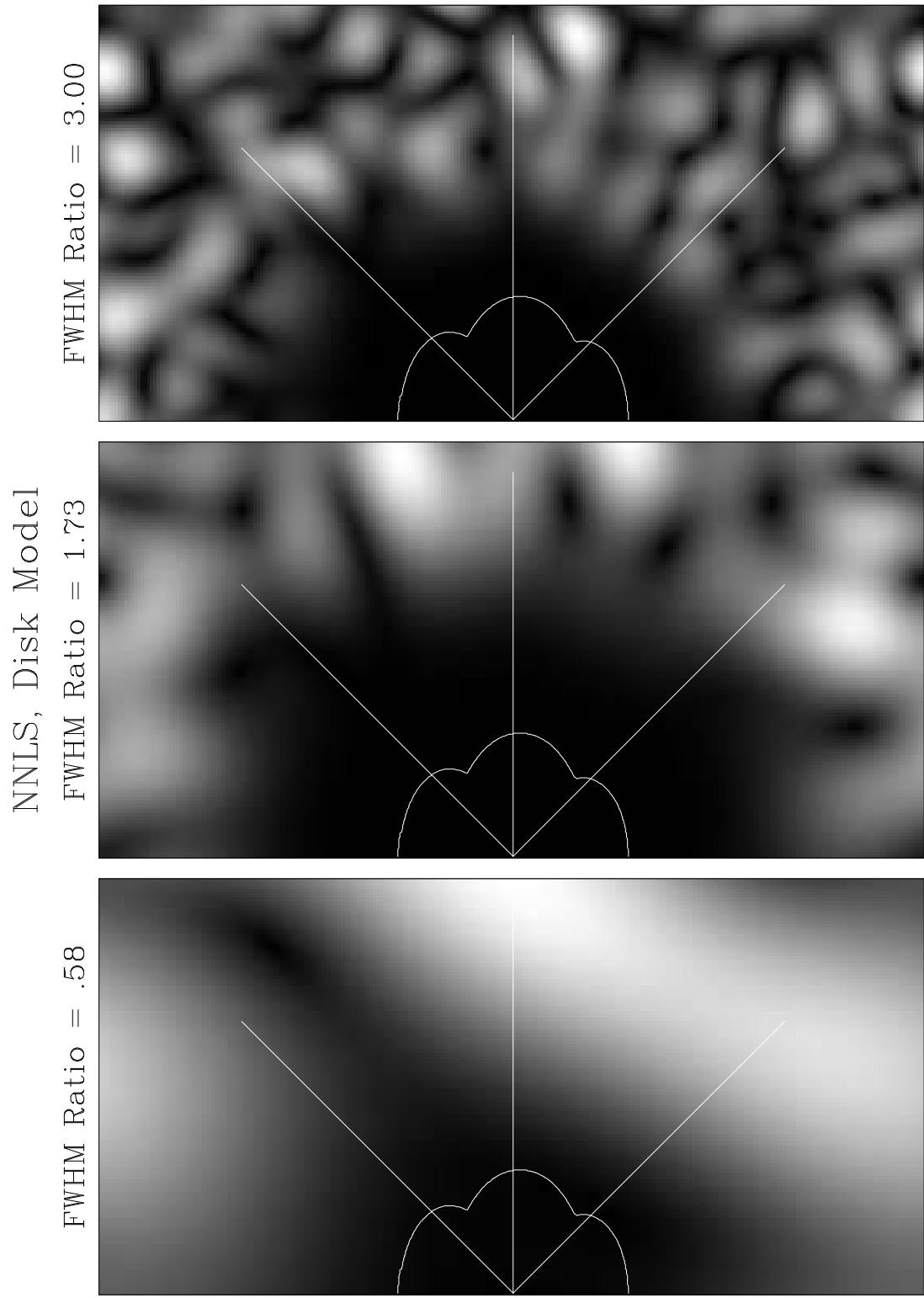


Figure 5.6a: Error Amplitude in the u - v Plane — NNLS, Disk Model

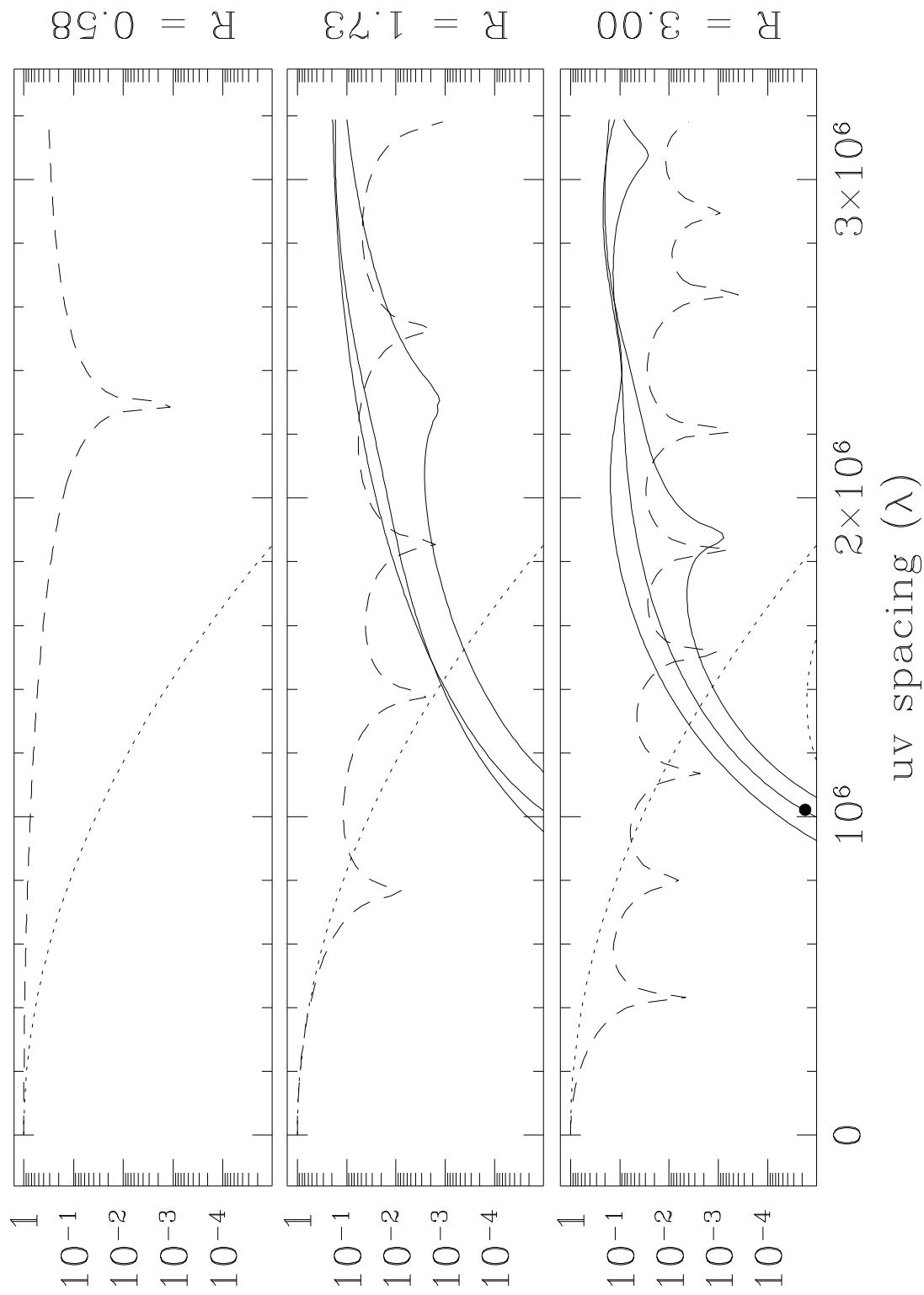


Figure 5.6b: Model Error Amplitude in the $u-v$ Plane — NNLSS, Disk Model

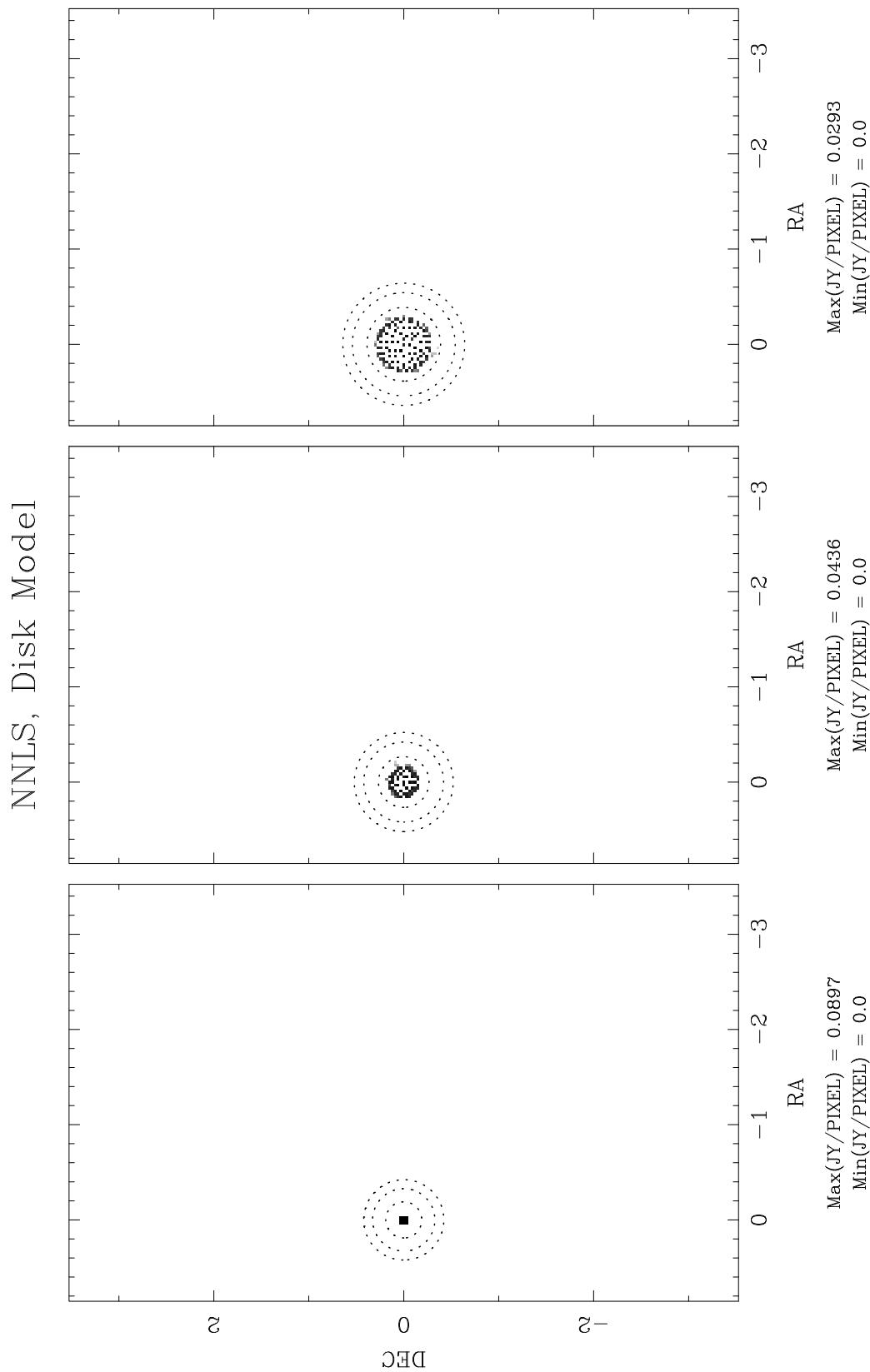


Figure 5.6c: Component Model in the Sky Plane — NNLS, Disk Model

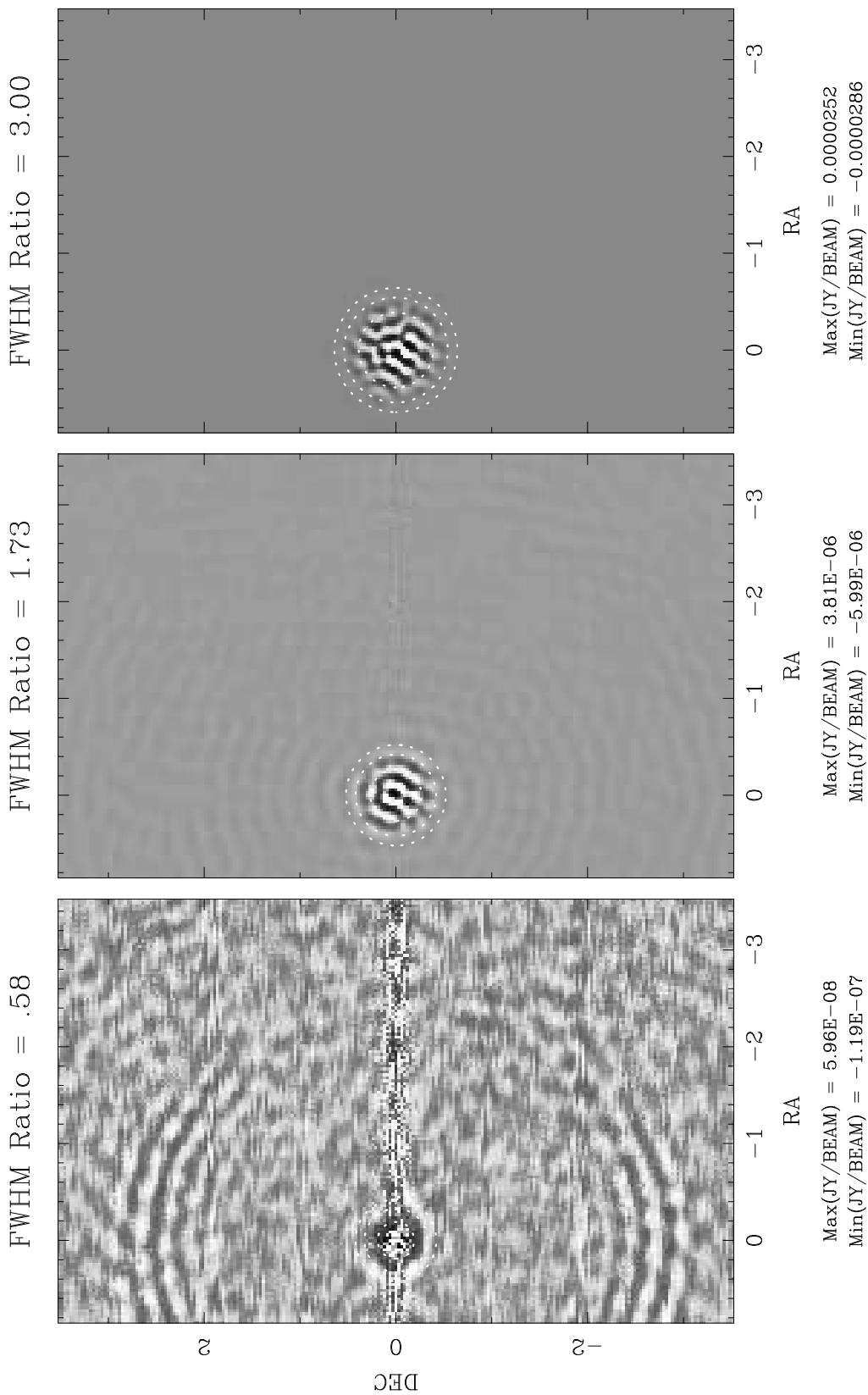


Figure 5.6d: Error Pattern in the Sky Plane — NNLS, Disk Model

| Gaussian – Component Model |

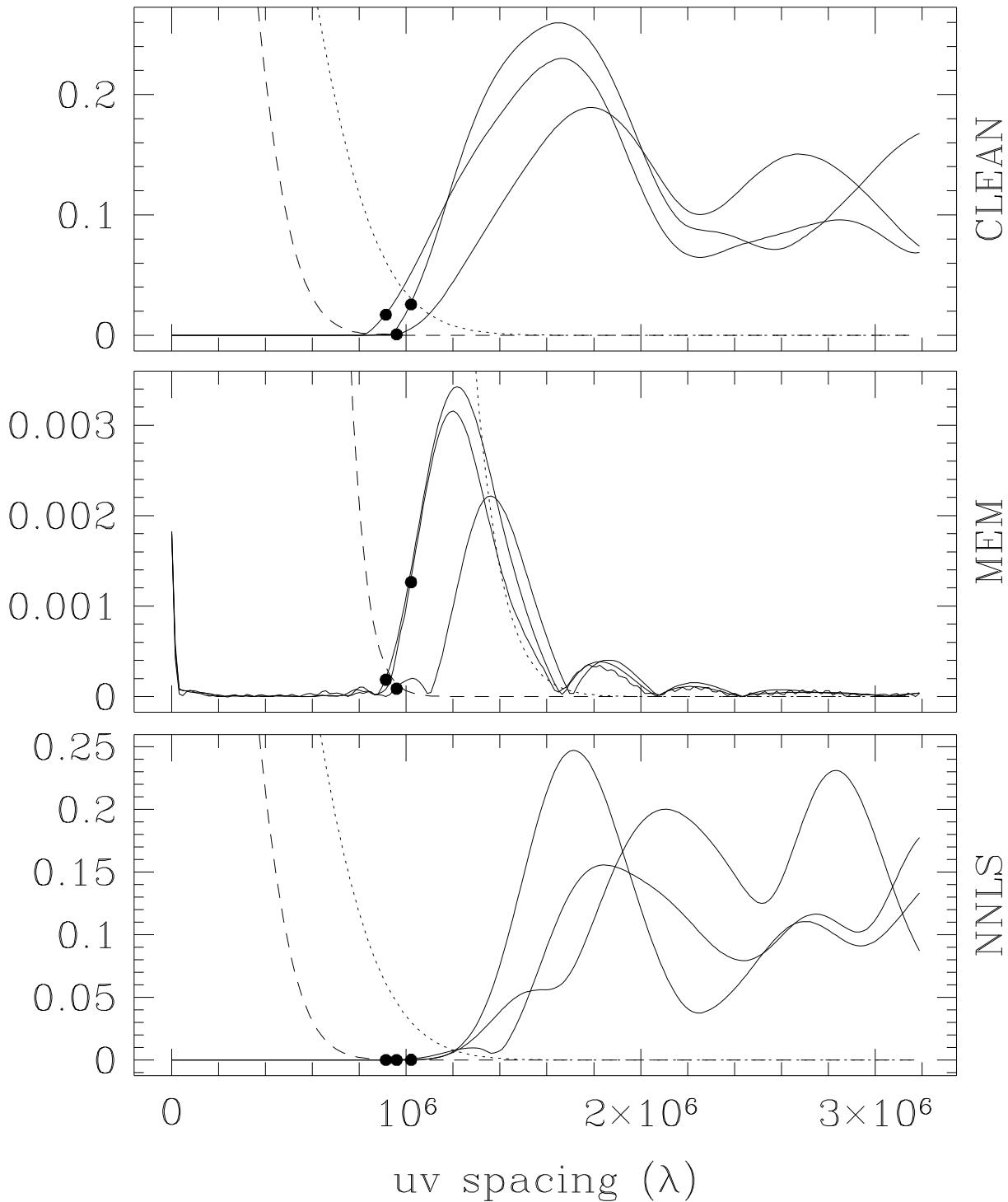


Figure 5.7: Model Error Amplitude in the Fourier Plane — linear algorithmic comparison. The source is the Gaussian $n = 2$ case for a model width 1.78 times the fitted beam width.

5.2.1 Results

If the deconvolution algorithm approximately zeros the residuals on a noiseless data set, then clearly the dominant reconstruction error must come from unsampled regions of the u - v plane. This is hardly new, but previous lore on the CLEAN algorithm has placed great emphasis on interpolation errors across holes in the sampling pattern. The most surprising thing about these error plots is how sharp the edge is at the u - v sampling envelope. Even when downweighted by the restoring beam, the dominant reconstruction errors almost invariably come at or just outside the sampling envelope. While these plots are from a well-sampled VLA coverage, the same statement is true even in a full track VLBA coverage experiment with considerable holes in the outer spacings such as that shown in Figure 3.12b. The errors are larger in the holes than in the well sampled areas, but even after smoothing both are dominated by the errors just outside the sampling.

Reconstruction errors in high-precision observations of simple sources are dominated by extrapolation errors, not interpolation errors.

The immediate corollary to this, of course, is that using a larger restoring beam will lead to smaller reconstruction errors. Again this is not new, but as we will see quantitatively in Figure 5.10, it is true to a remarkable extent. Since reconstruction errors quickly rise to a significant level outside the sampling envelope, the errors in the smoothed model are strongly influenced by the location of the errors relative to the restoring beam transform. The worst errors occur closer in, and often nestle into a “corner” of the sampling pattern.

More algorithm specific observations about the error morphology are

- CLEAN has the worst extrapolation properties of the three major algorithms — the sharpest edge at the sampling envelope — followed by NNLS and MEM for the Gaussian cases. In the disk cases the ordering changes to CLEAN, MEM, NNLS in decreasing order of model error sharpness at the sampling envelope.
- CLEAN and NNLS will place power at high spatial frequencies, whether it is there or not, and MEM will avoid placing power there, whether it is present or not. This is an obvious consequence of the smoothness criterion or lack of it built into the algorithms.

Simply putting spatial power into a region where the true source has power does not guarantee the reconstruction there will be correct; the phase may still be wrong. But not putting power there will certainly be wrong. By contrast, not placing power in a region where the source has none is guaranteed to be correct. This dichotomy tends to favor MEM for reconstruction of extended smooth objects.

- The error patterns produced by CLEAN have a random structure, though a characteristic scale size. NNLS errors are somewhat more structured, with a striking periodic speckle appearing on smooth emission, which is the transform of the regular spacing

of model components. The most regular error patterns are produced by MEM, in particular on the Disk models where the outer structure in the error pattern is essentially the pure model.

- CLEAN has a striking inverse correlation between the scale size of the mottling pattern on its error pattern and the source size. It is not related in detail to the structure of the model transform.
- NNLS does a superb job of modelling the source inside the sampling envelope in the u - v plane, CLEAN does a fair job, and MEM the worst of the major algorithms.
- NNLS has no problems with the sharp-edged sources, CLEAN a bit more, and MEM has real problems with them.
- The CLEAN error morphology in the image plane is nearly constant for $n < 0$, varying only in absolute scale. It is still similar at $n = 2$.
- CLEAN has a ringing structure in the placement of its components that will be investigated in more detail in Section 6.2.
- NNLS tends to place components very regularly on regions of smooth emission. This can be seen most clearly by the strong peaks in the u - v plane model error plots.
- NNLS error patterns are strongly confined to the support of the source. Its performance compared to other algorithms gets better if one considers the off-source errors as opposed to errors anywhere in the image.
- CLEAN is doing better than MEM at the *low* spatial frequency reconstruction, and for all but the smallest and sharpest sources, MEM is doing better than CLEAN at the *highest* spatial frequencies. Notice that this is exactly backwards to what one might expect from the statements “MEM is good at smooth emission, and CLEAN is good at compact emission.”

Acting on the observation that the CLEAN reconstruction error very quickly rises to nearly a constant level immediately outside the sampling envelope, an effort was made to model the CLEAN errors by an azimuthally-symmetric Heaviside step function in radius, with an amplitude produced by an incoherent sum of a number of components proportional to the source area. A correlation was sought between the fraction of the restoring beam beyond the sampling envelope times the modeled error amplitude. This was not successful, though it is not known whether the problem was that the model was not detailed enough, or if the correlation structure of the amplitude errors was causing the problems. Image plane modelling is an effective way to estimate the error due to deconvolution on simple sources, so the Fourier plane approach of modelling the errors was not pursued.

5.3 Residual Behavior

The previous discussion largely assumed that the residuals of the deconvolution process were ignorable, that is, that the error in the smoothed model was roughly equivalent to the error in the reconstructed image. This is only approximately true. In detail, the algorithms differ greatly in how well they can (or attempt to) zero the residuals. We now consider all 6 of the resolution-test algorithms, adding Maximum Emptiness, SDI CLEAN, and Gerchberg-Saxton-Papoulis.

First, note that Maximum Entropy and Maximum Emptiness do not try to zero the residuals at all — they try to fit the data only within the thermal noise, and the level of agreement is regarded as a free parameter to be adjusted by the user. The minimization of their particular criterion will almost certainly mandate a solution that does not zero the residuals, and if started at such a solution they will move away from it. The 4 remaining algorithms will zero the residuals in the sense that if started with an allowed model that exactly fits the data, succeeding iterations would remain at that point. In a more practical sense, however, the convergence behavior of SDI CLEAN and GSP is sufficiently slow in the high-precision cases that the residuals are never completely zeroed. CLEAN converges somewhat faster than SDI CLEAN on a “per component” basis, though not necessarily on a “per machine cycle” basis. Given sufficient time it will reduce the residuals to zero and the number of components required is modest enough that at least some small sources can be zeroed of machine precision. Of all the algorithms, NNLS is generally the best at zeroing the residuals, though of course it can be used only for compact sources. This probably is related to its usefulness in self-calibration.

Figure 5.8 presents the image-plane residual level for the basic resolution-test series. The PSF used is the Full Track VLA PSF from the previous section, so the corresponding points of these plots are exactly the same cases as presented in Section 5.2.

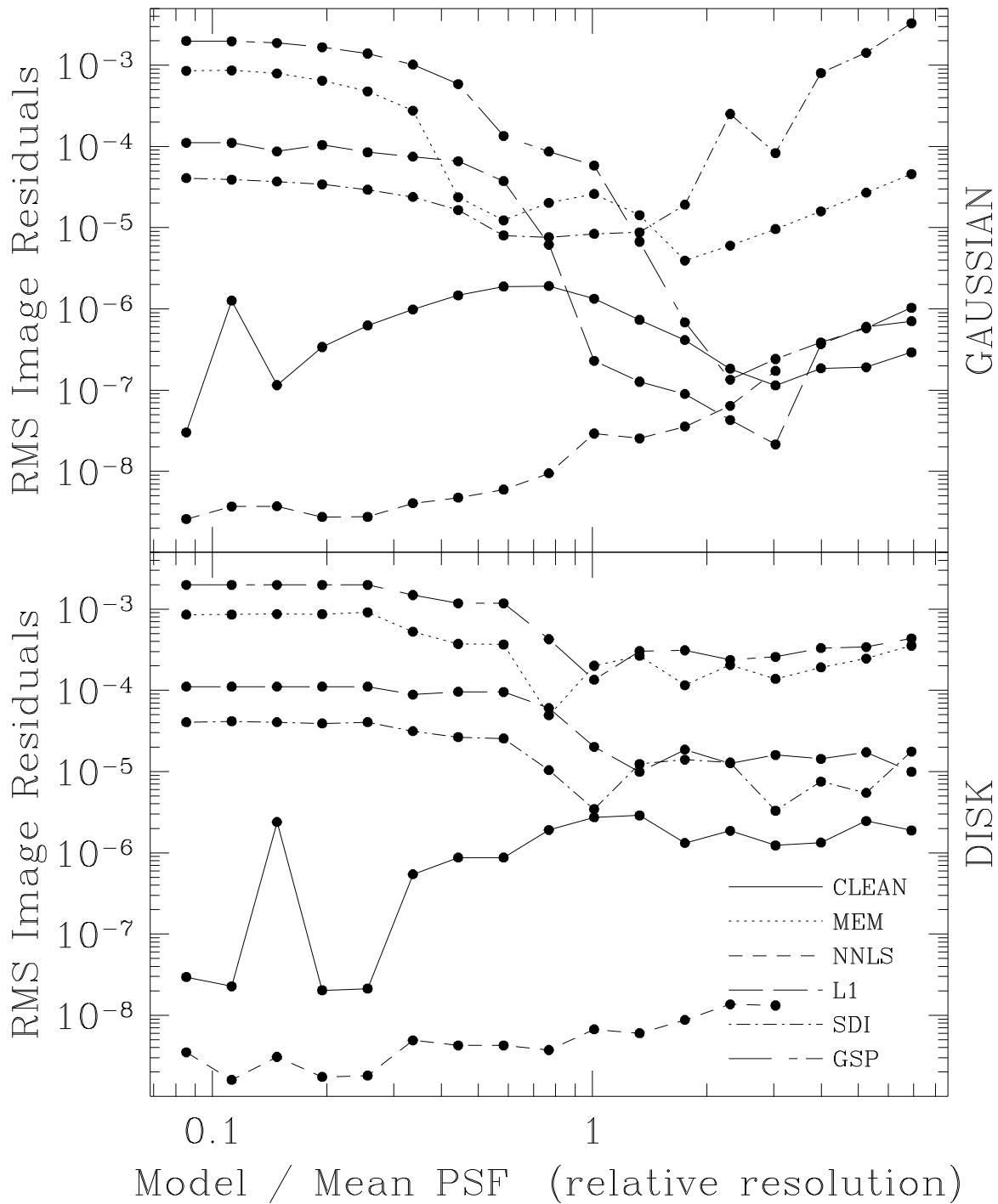


Figure 5.8: RMS residuals in the image plane.

5.4 Algorithmic Cross Comparison

In this section we use only the Full Track VLA point spread function and examine how well the different algorithms do relative to one another. Figure 5.9 presents the error statistics against source size for all six algorithms, restoring at the near-nominal .2'' beamwidth. Figure 5.10 presents the same quantities, but examines only a single source size and varies instead the restoring beamwidth. These two figures form slices though the center of a full two-dimensional parameter space explored. The ASCII data files necessary to produce plots like these for the full range of parameters are available in the electronic supplement to the dissertation, as described in Appendix H. In these plots, a solid circle is a positive reconstruction error, (too much flux at a point), and an open circle is a negative error.

- Deconvolution errors can be quite significant, even for a well sampled VLA observation. These errors often look like calibration errors, and have probably been misdiagnosed in the past, along with whatever calibration errors are actually present.
- Some algorithms almost always make their errors in one direction or the other, while others are nearly random errors.
- CLEAN errors are dominated by $u-v$ plane extrapolation, not holes in the sampling pattern or lack of zero spacings. This makes CLEAN based algorithms very amenable to smoothing.
- Even very mild resolution produces significant CLEAN deconvolution errors. The only thing that is a point to CLEAN is a real point source. A source with a width as much as 20% of the fitted beamwidth behave like an extended source when CLEANed.
- MEM errors have a significant zero spacing component. MEM images are often significantly better than the CLEAN images at nominal resolution, but cannot be improved by post smoothing. Fractional RMS error may actually get worse with smoothing.
- The source itself masks the on-source CLEAN errors. This leads to a strong peak in “off-source errors” as a function of source size when the source size is slightly less than the synthesized beamwidth.
- The algorithms which make their dominant errors outside the $u-v$ sampling envelope are the ones with steep slopes in Figure 5.10.
- Either CLEAN or MEM is more effective than SDI CLEAN for nearly all size scales
- GSP and L1 can produce very good reconstructions on some sources, but they behave quite badly on others. In general, these algorithms were not found very robust to support constraints.

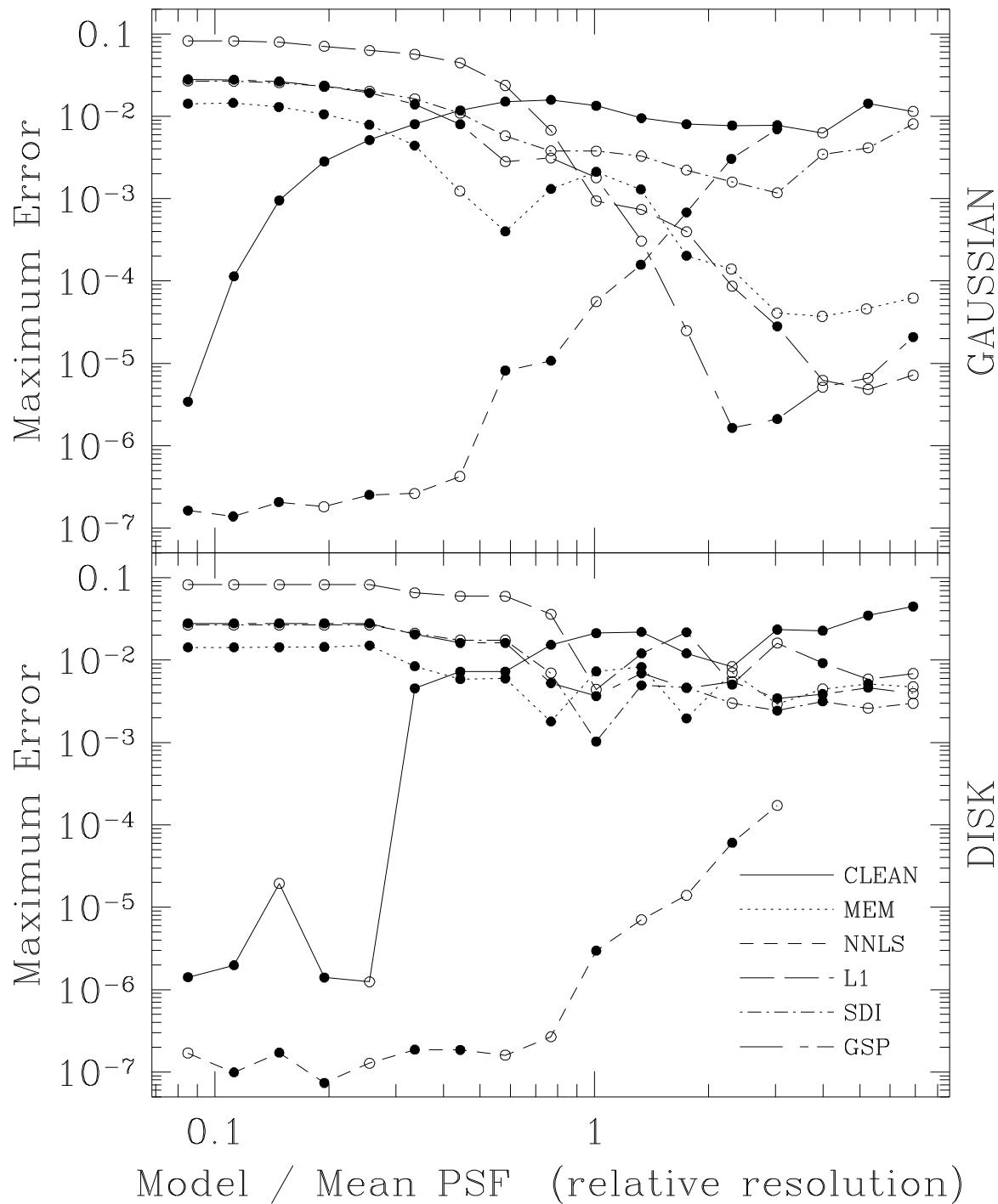


Figure 5.9a: Deconvolution errors against source size, Algorithmic comparison. Maximum Error in the smoothed reconstruction. Reconvolving size was $.200''$, and the fitted Gaussian FWHM was $.194''$.

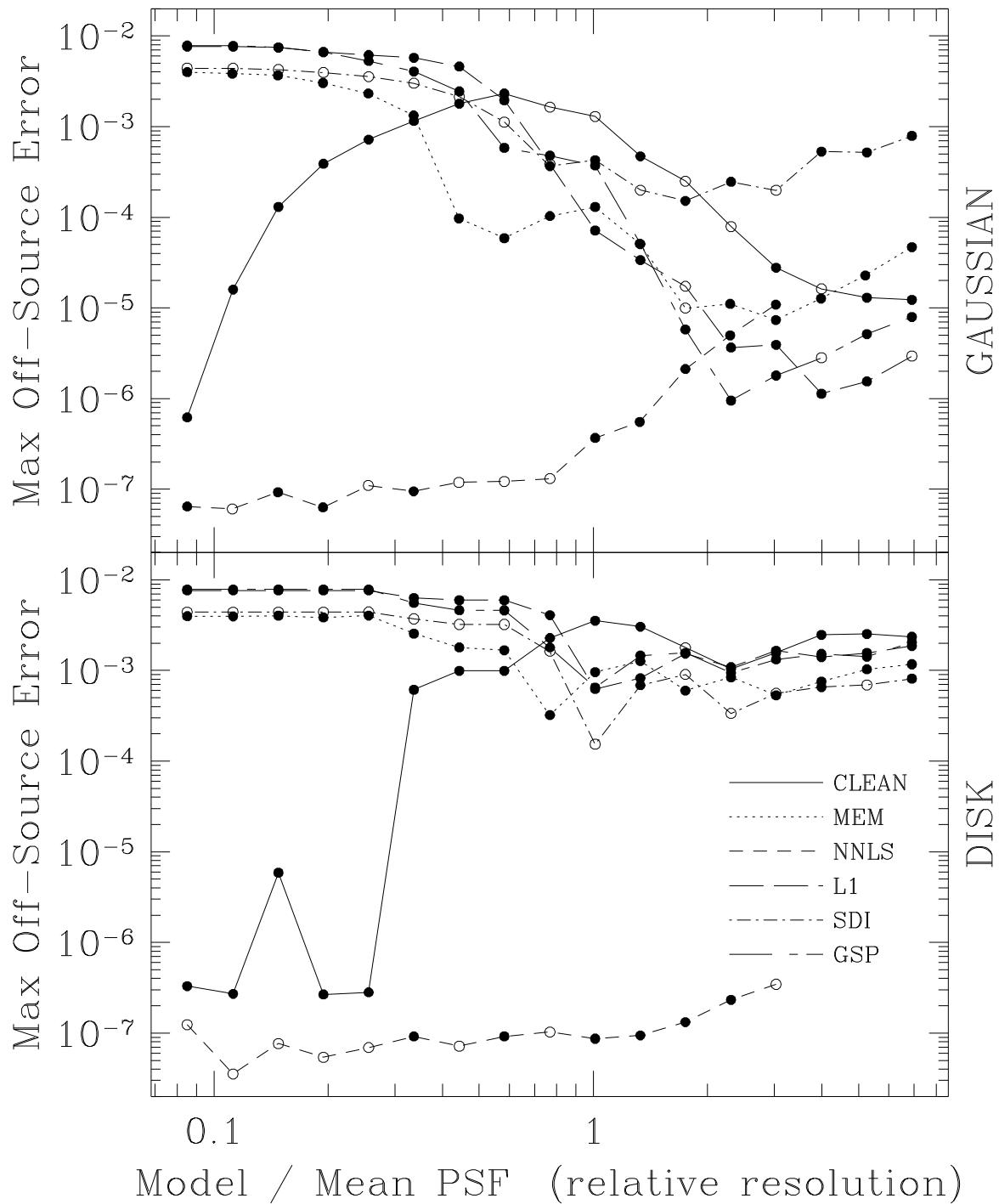


Figure 5.9b: Same as Figure 5.9a, but maximum error off the support of the smoothed source.

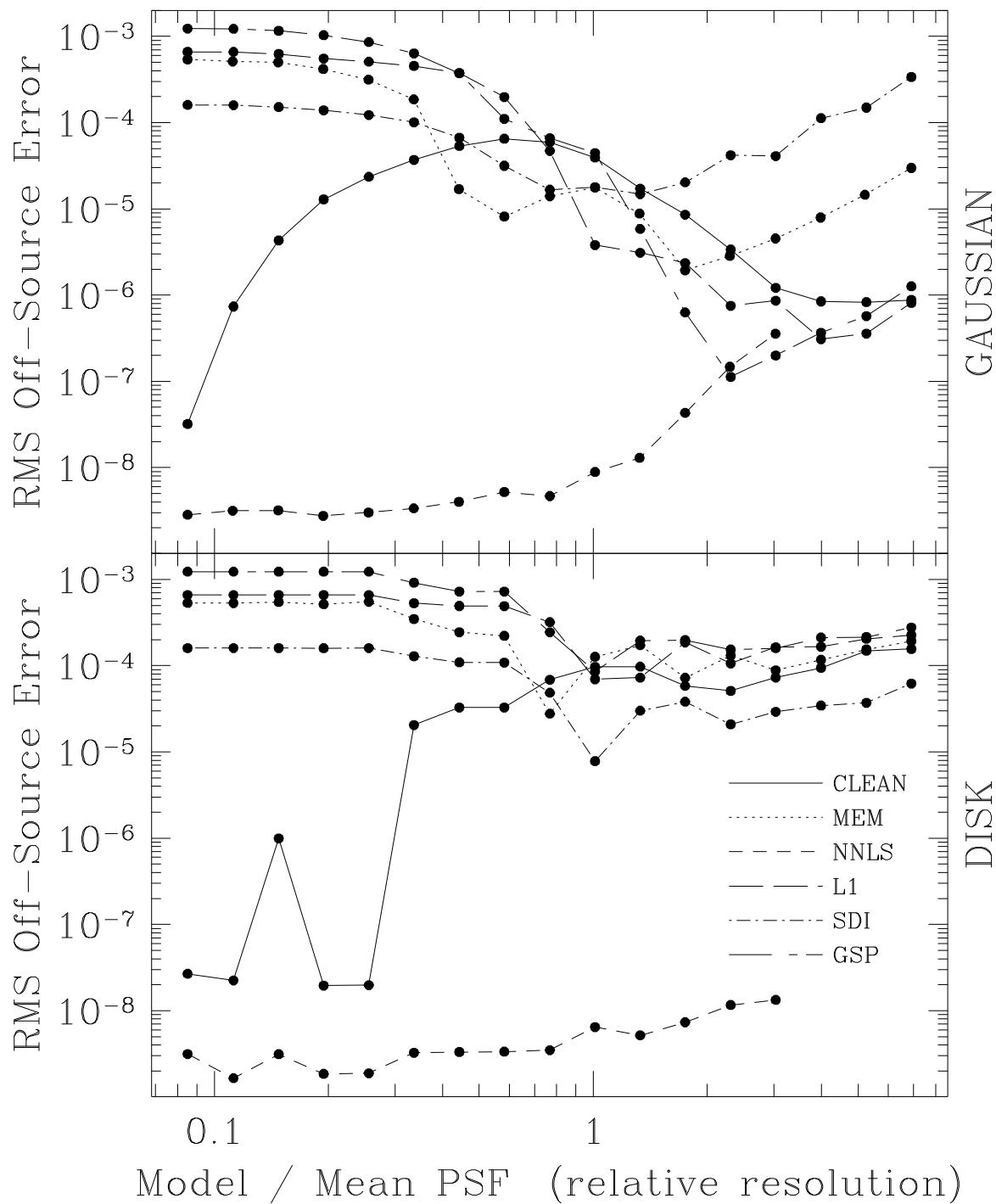


Figure 5.9c: Same as Figure 5.9a, but root mean square error off the support of the smoothed source.

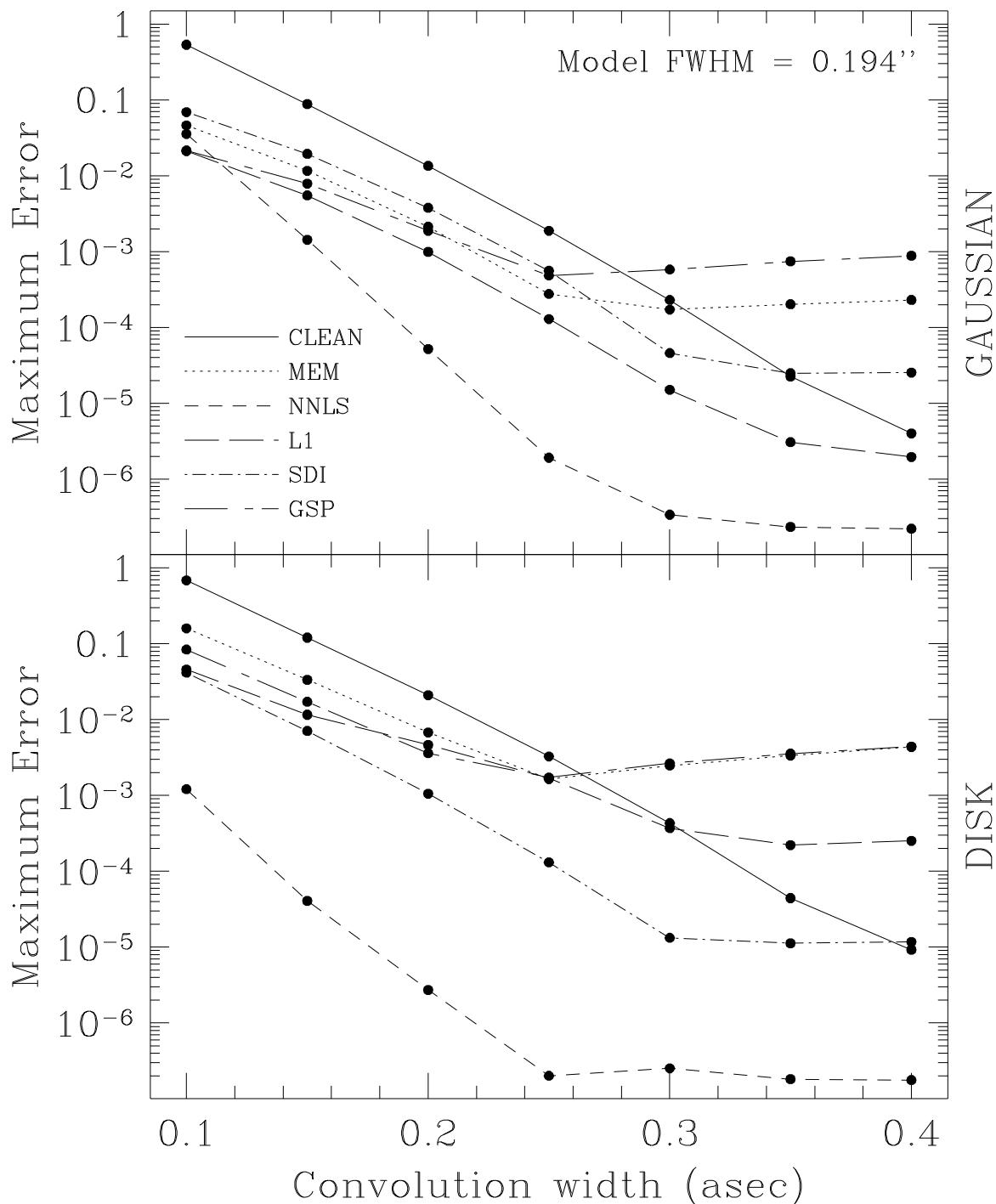


Figure 5.10a: Deconvolution errors against reconvolution size, Algorithmic comparison. Maximum Error in the smoothed reconstruction. Source FWHM was $.196''$, the size of the fitted beam.

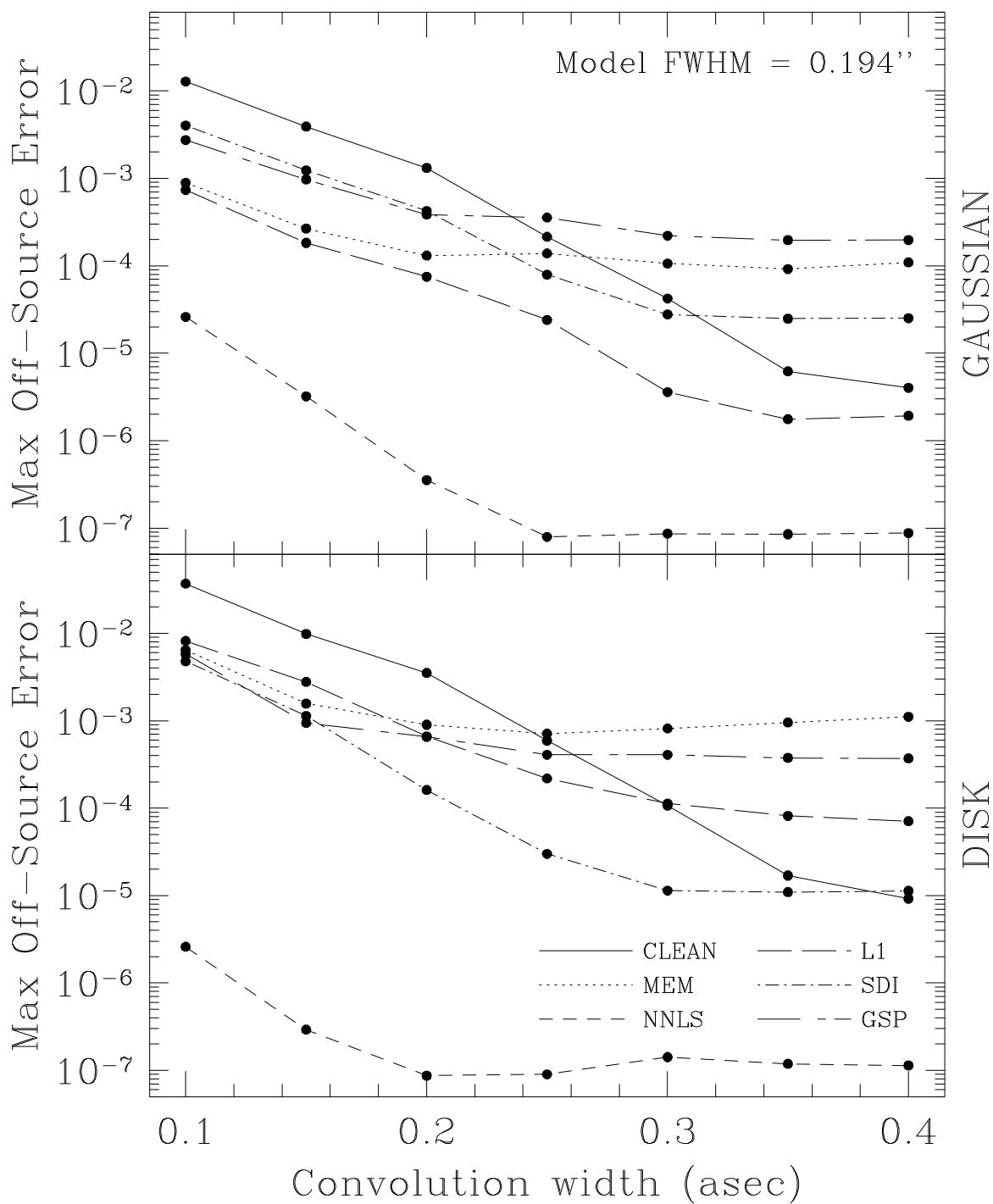


Figure 5.10b: Same as Figure 5.10a, but maximum error off the support of the smoothed source.

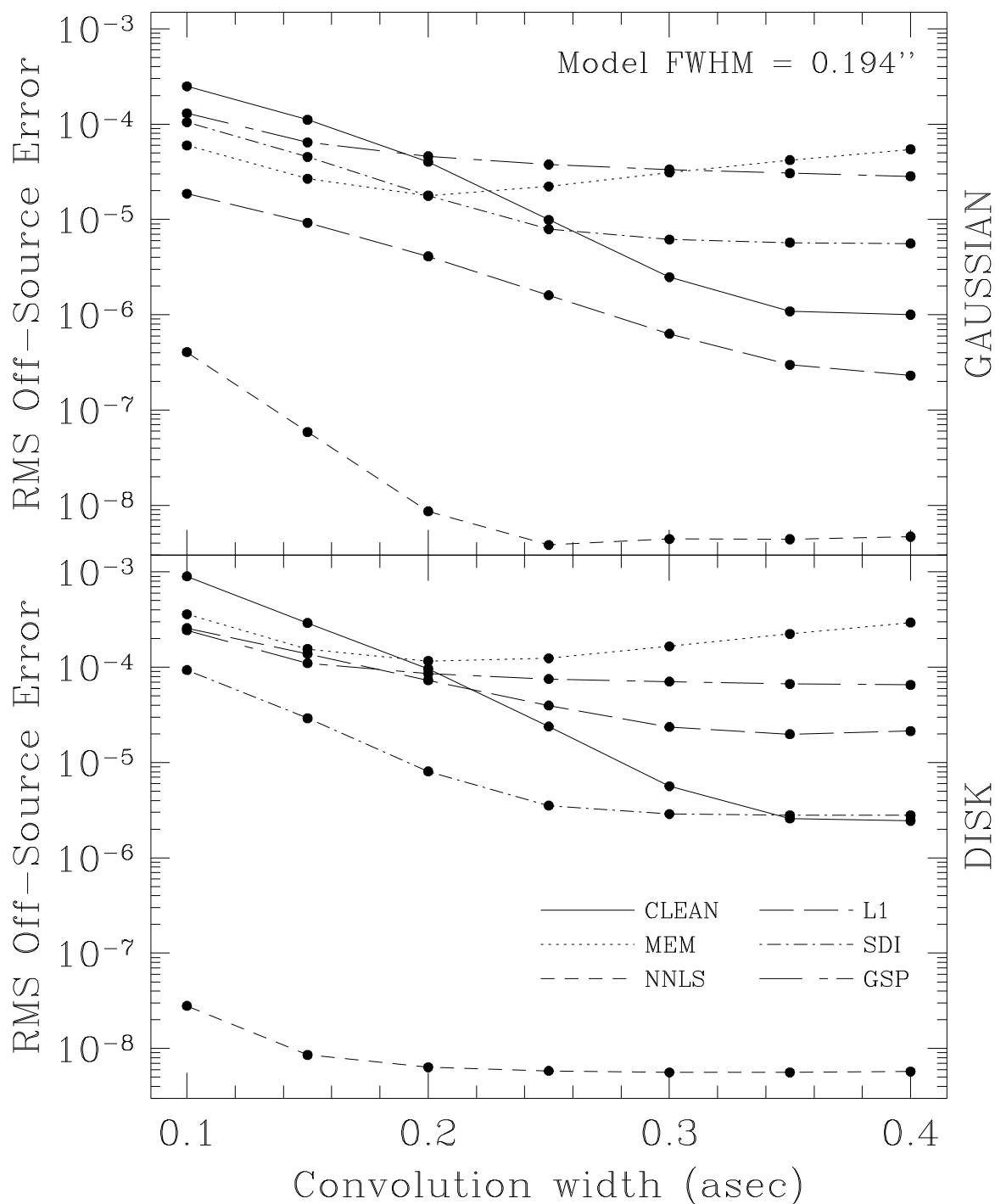


Figure 5.10c: Same as Figure 5.10a, but root mean square error off the support of the smoothed source.

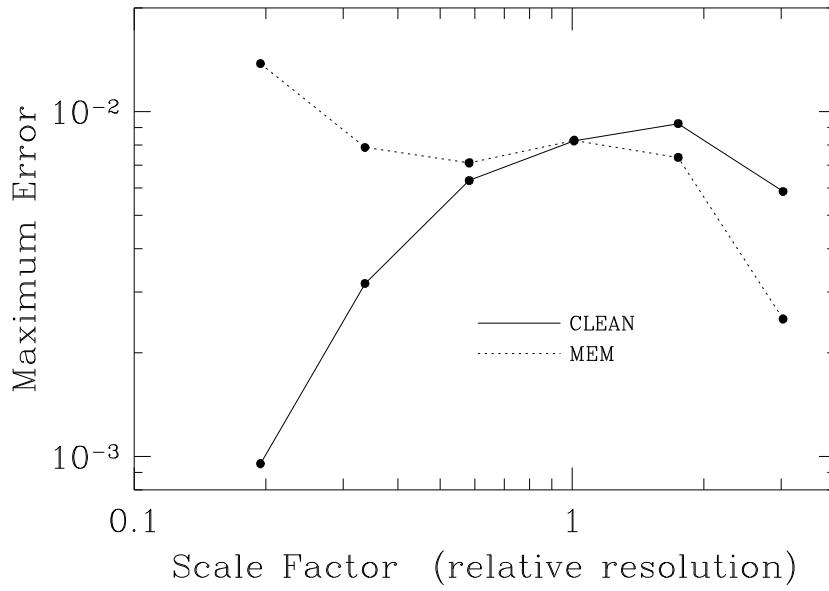


Figure 5.11a: Deconvolution errors against source size. Unit scaling corresponds the model in Table 3.8.

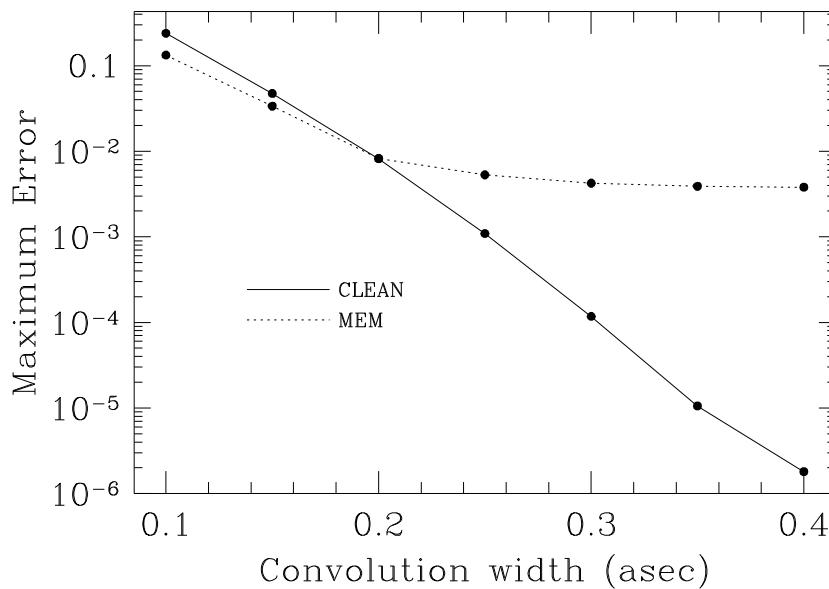


Figure 5.11b: Same model as in Figure 5.11a at Scale = 0.

5.5 The Effect of Fourier Plane Coverage

We now restrict our attention to the three major algorithms once again, and ask how well they can recover the source when the u - v coverage is varied significantly. The simulations use the same Gaussian and Disk sources as before, and the uniformly weighted point spread functions from Figure 3.13 (VLA Full Tracks (3C48), referred to here as VLA Full Tracks), Figure 3.15 (VLA 30 Point Snapshot) and Figure 3.20 (VLBA Full Tracks).

In addition to the usual samplings, we also include the limiting case of a “perfect” VLA observations, where the transfer function has been determined by simply tracing the outline of the VLA sampling pattern and setting all interior points to unity. The Mid Tracks sampling was intended to loosely represent the practice of interleaving many different sources in survey style observations. The two special samplings are given in Figure 5.12 (VLA Mid Tracks) and Figure 5.13 (VLA Perfect Coverage).

- The most surprising effect is how little effect there is. Only the snapshot coverage makes much difference. All the rest are pretty much the same. One should be cautious about extrapolating this result to more complicated sources!
- MEM is the most sensitive of the three algorithms to u - v coverage, followed by CLEAN which is much better. (The three largest snapshot cases diverged when run to the full iteration limit with CLEAN. The points shown were limited to reasonable iteration values manually.) NNLS is nearly immune to variation in u - v coverage for these cases.

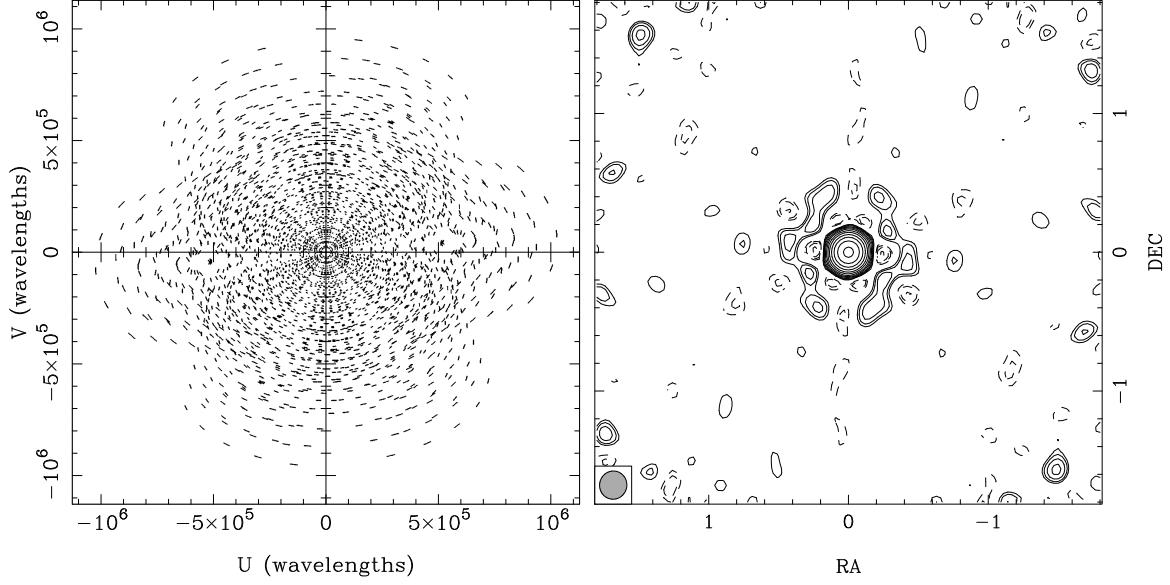


Figure 5.12: Mid Tracks u - v coverage and PSF. This is one hour of u - v data, divided into 5 12 minute scans centered at hour angles of $-2, -1, 0, 1$, and 2 hours. Integration time is 30 seconds, and the declination is 35° . Minimum contour is 2%.

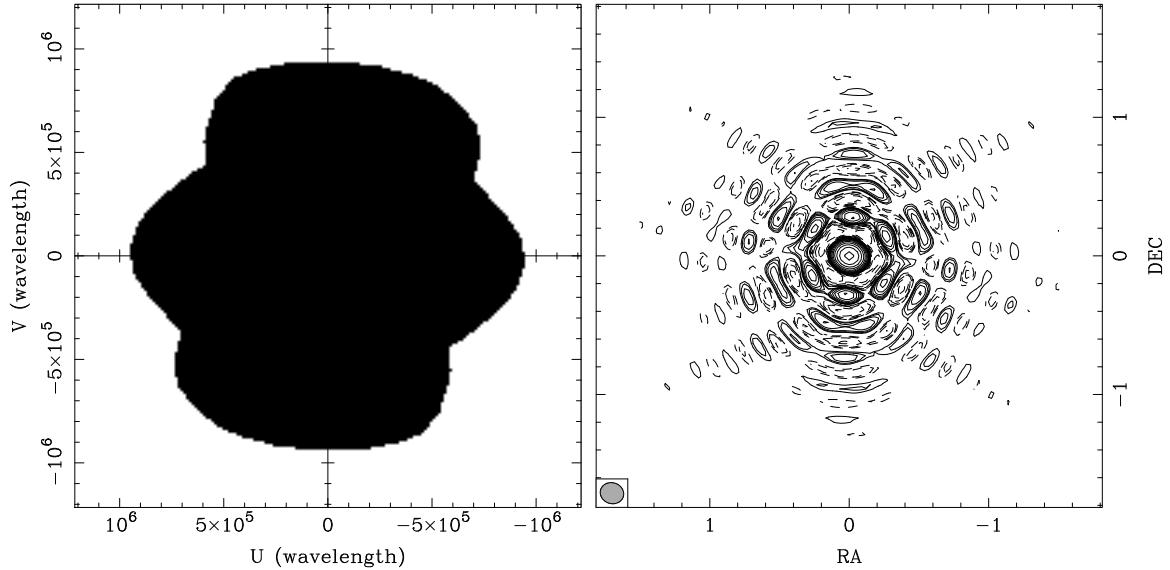
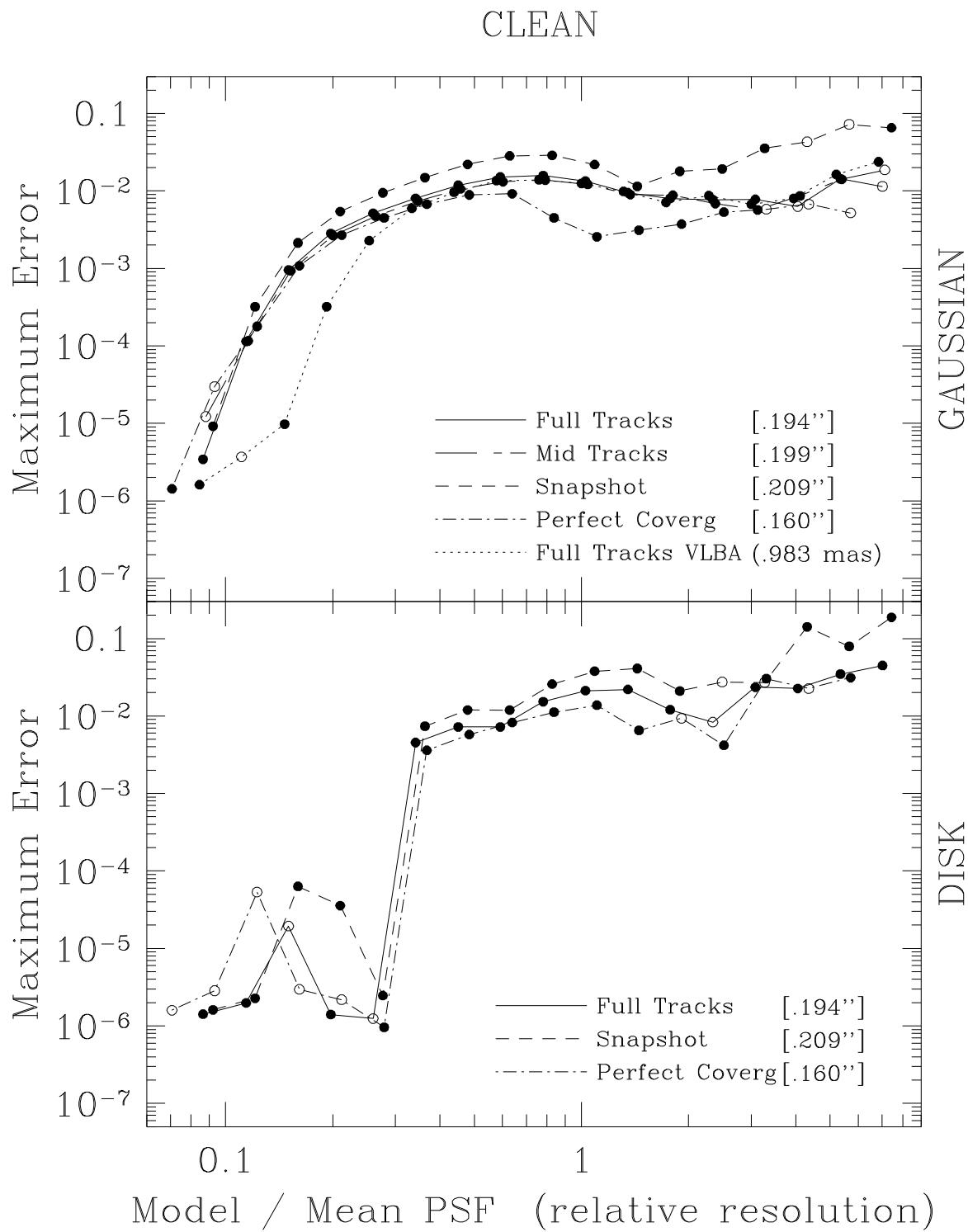
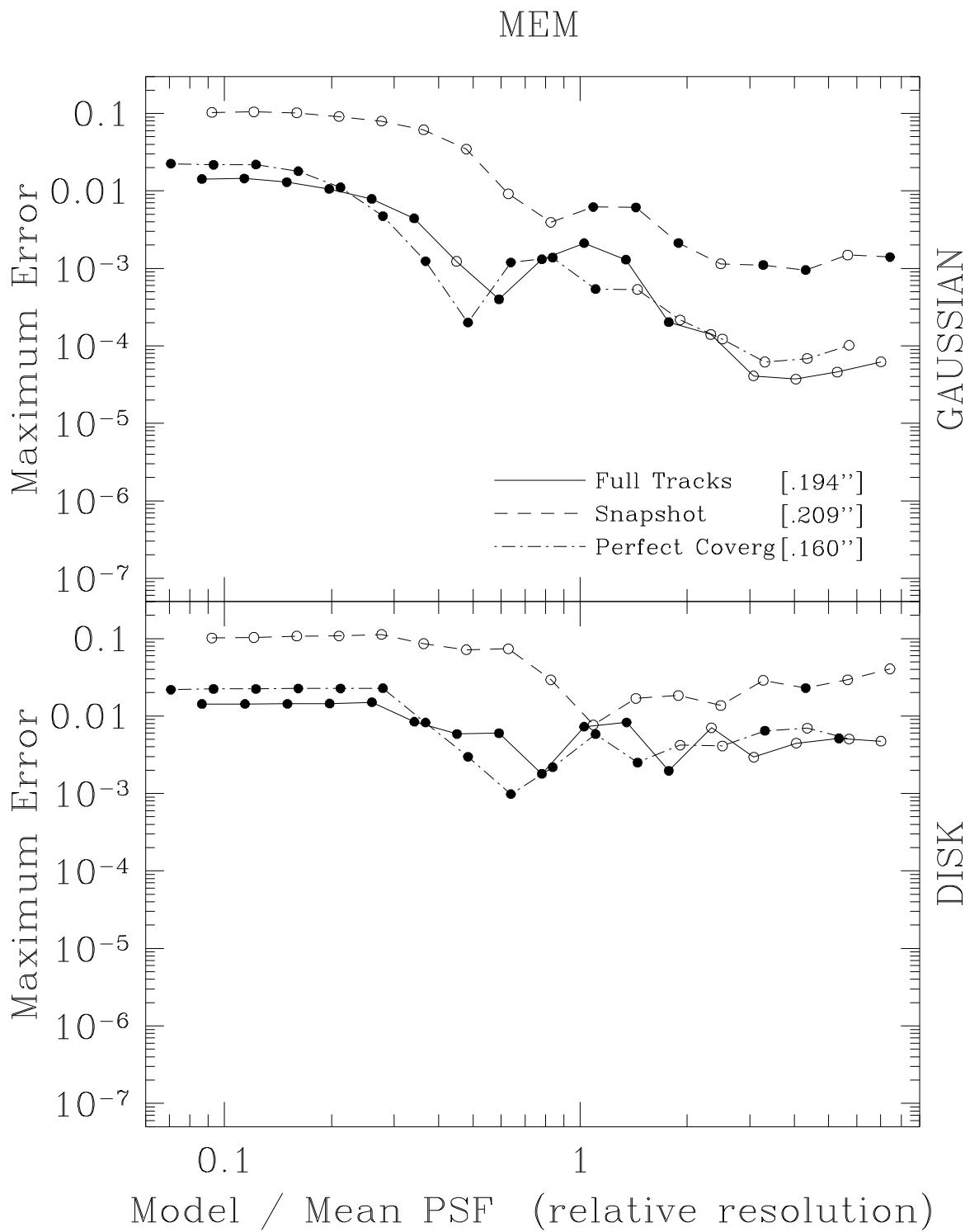
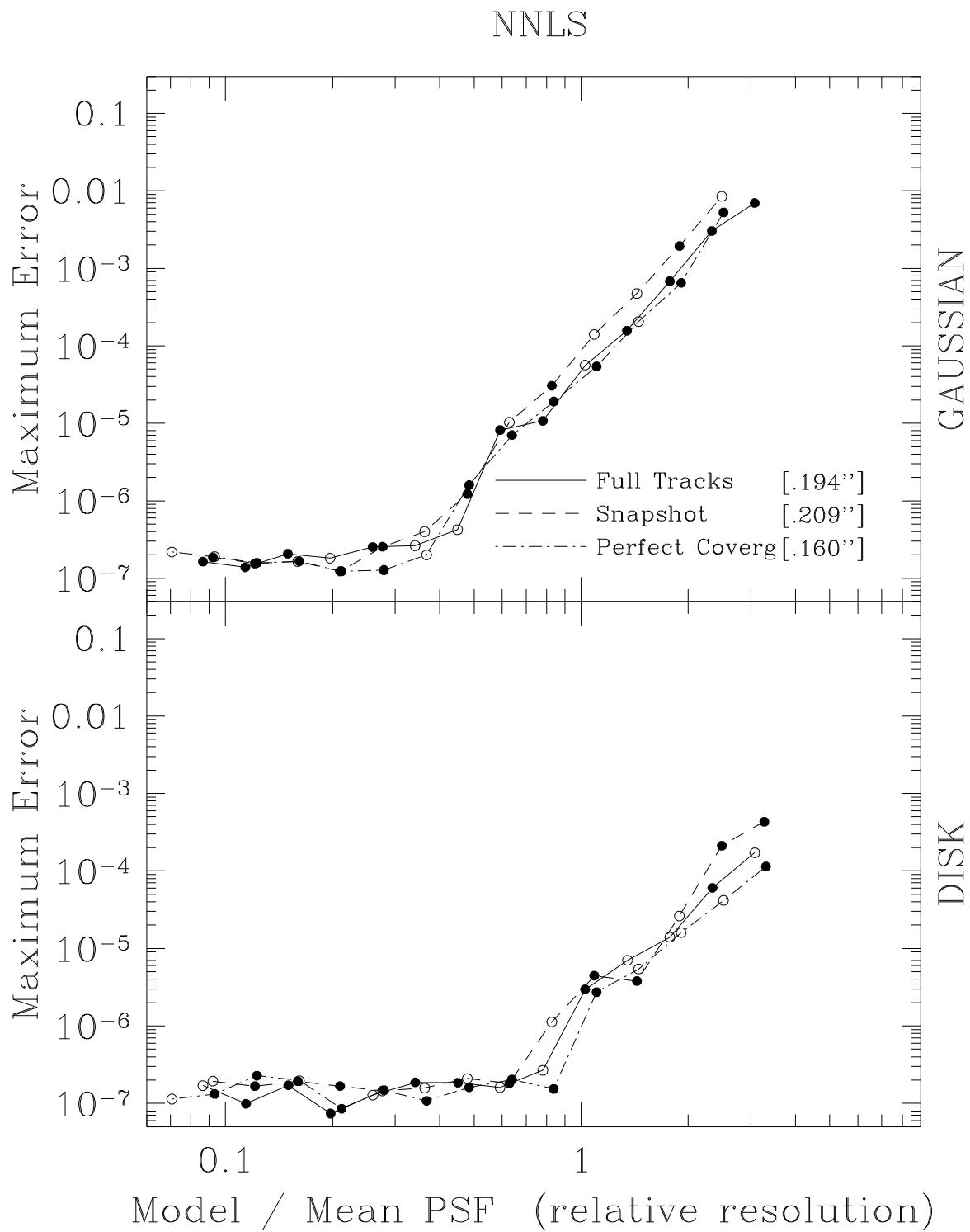


Figure 5.13: Perfect VLA u - v coverage and PSF. This PSF was generated by simply tracing the outline of the VLA sampling pattern, and setting each interior pixel in the gridded transform plane to unity. Declination is 35° . Minimum contour is 1%

Figure 5.14a: *u-v* Coverage and Deconvolution Errors — CLEAN

Figure 5.14b: $u-v$ Coverage and Deconvolution Errors — MEM

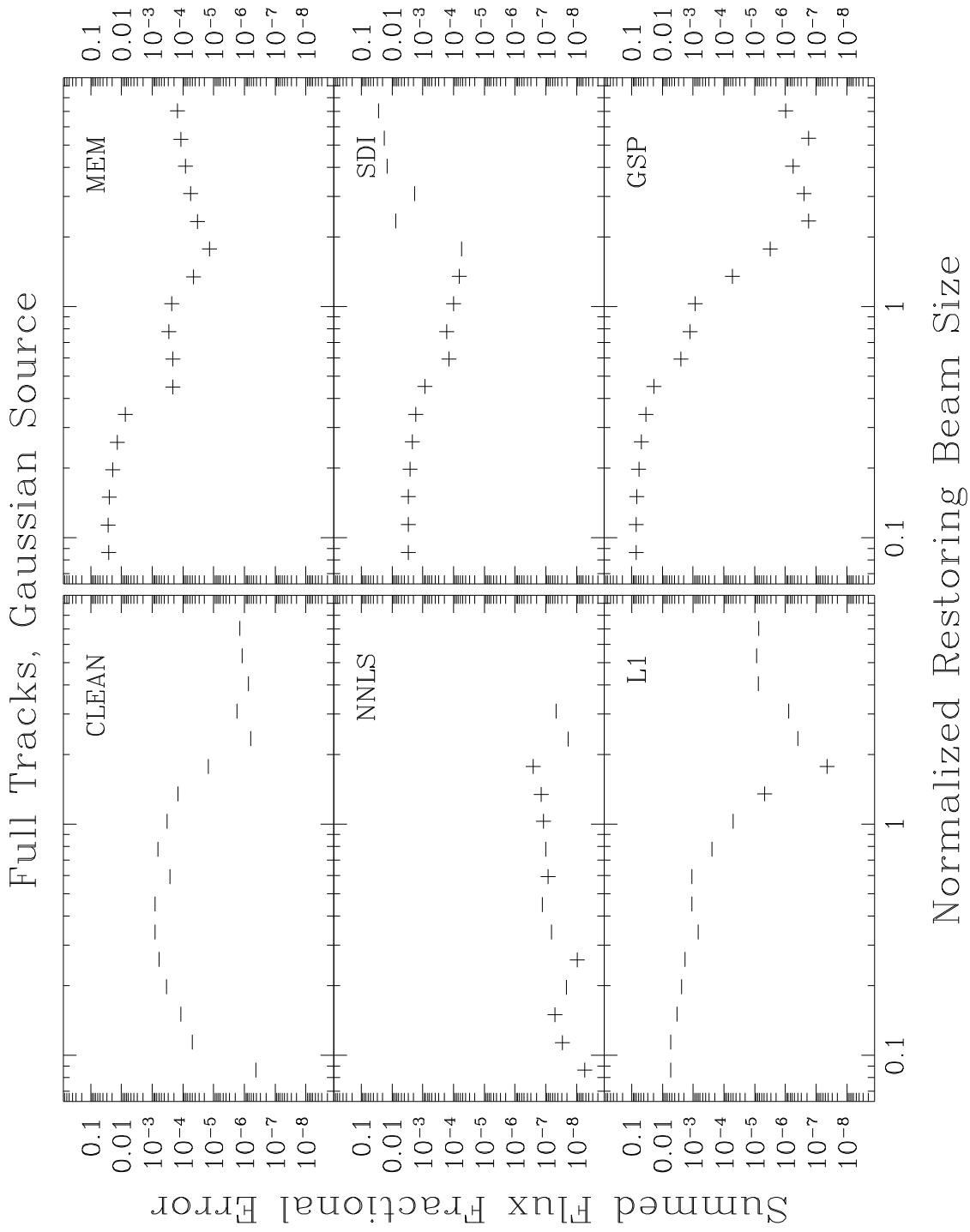
Figure 5.14c: $u-v$ Coverage and Deconvolution Errors — NNLS

5.6 Flux estimation

Flux estimation can be a tricky business, and given the importance of photometric observations, there has not been enough work done on the subject. There are many different factors that can influence the estimated total flux in a source, namely the shape of the source, the shape of the beam, the noise level of the observation, the calibration errors in the observation, and the interaction of the deconvolution algorithm with all of these.

These plots attempt to quantify just the contribution of the deconvolution algorithms on perfect data, in the limit where the algorithms have essentially converged. These are the same resolution-test deconvolution as before, but now the flux has been determined by adding up restored image pixels in the ‘on-source’ window and dividing by the restoring beam area. Only two beams have been used here, the full-tracks uniform beam and the snapshot uniform beam. This is a significant omission, and more work should be done here. In particular, these plots should be repeated with a best-fit Gaussian robustly weighted beam.

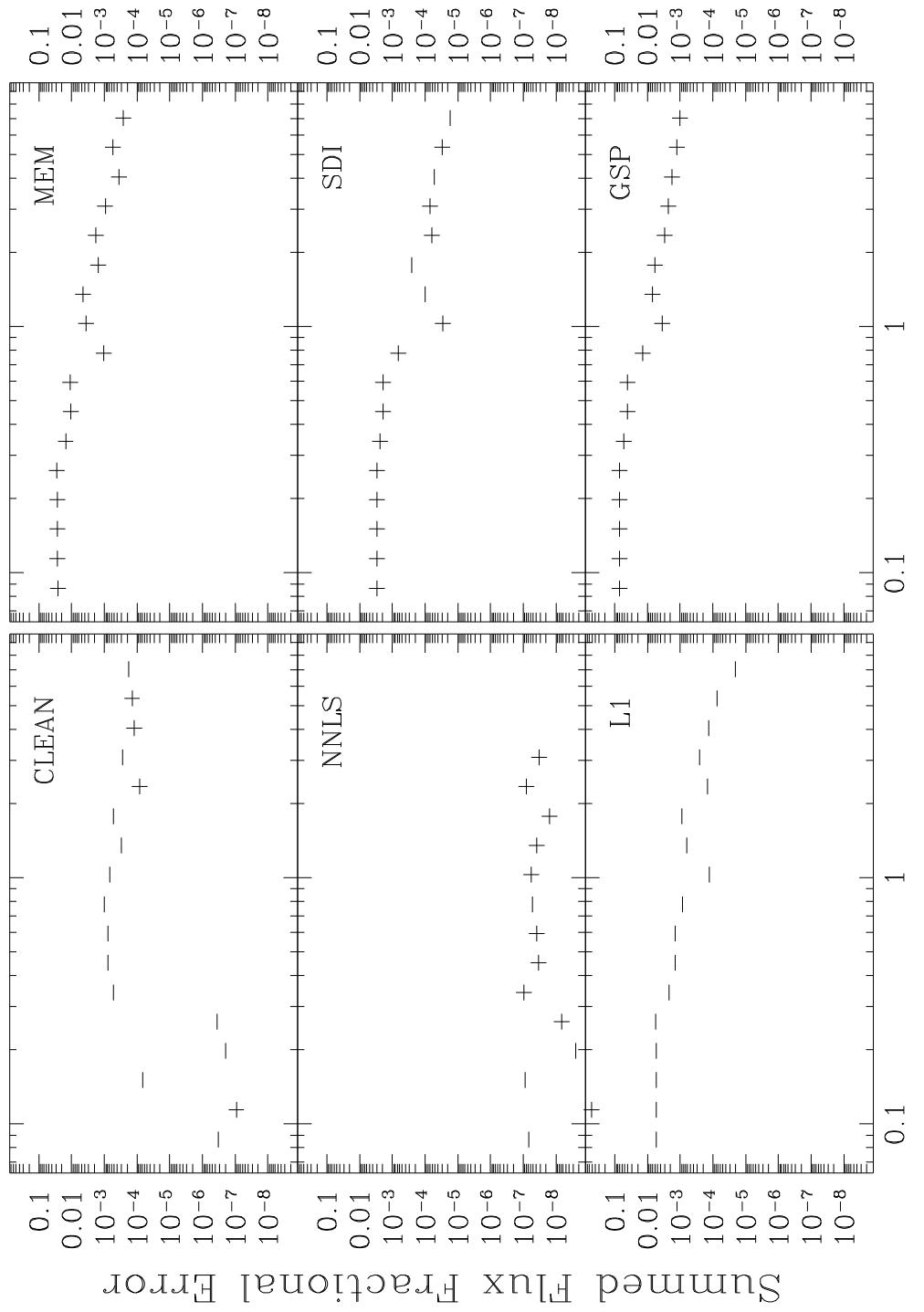
- NNLS is the best of the algorithms examined for flux estimation in terms of the error magnitude. In addition, it seems to be the only algorithm that is unbiased in the flux estimated. All the others estimate systematically too high or too low.
- CLEAN is the next most useful algorithm for flux estimation, and is the only choice for larger sources, possibly excepting GSP.



Normalized Restoring Beam Size

Figure 5.15a: Flux Estimation Errors, Gaussian Source, Full Tracks

Full Tracks, Disk Source



Normalized Restoring Beam Size

Figure 5.15b: Flux Estimation Errors, Gaussian Source, Full Tracks

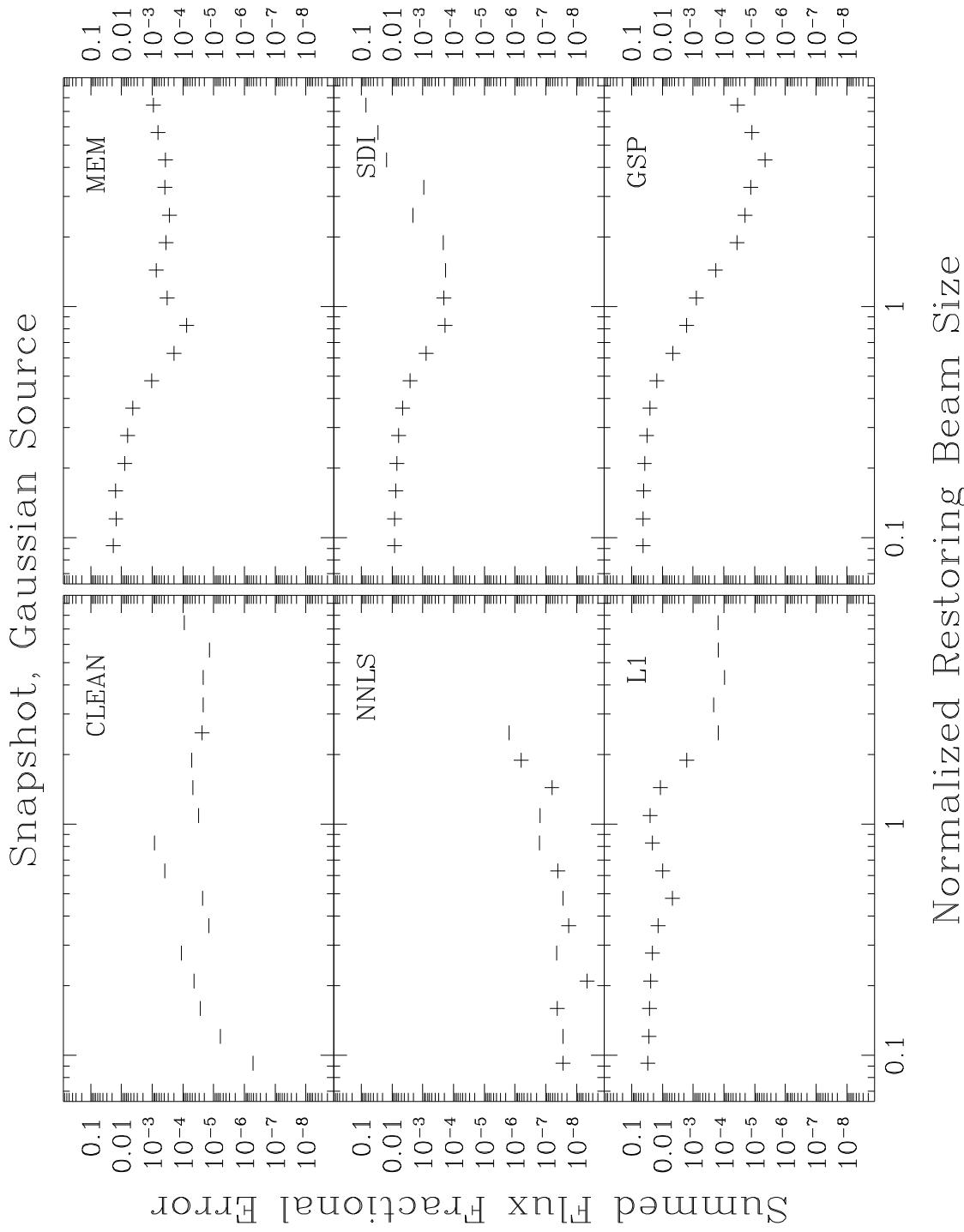


Figure 5.15c: Flux Estimation Errors, Gaussian Source, Snapshot Coverage

Snapshot, Disk Source

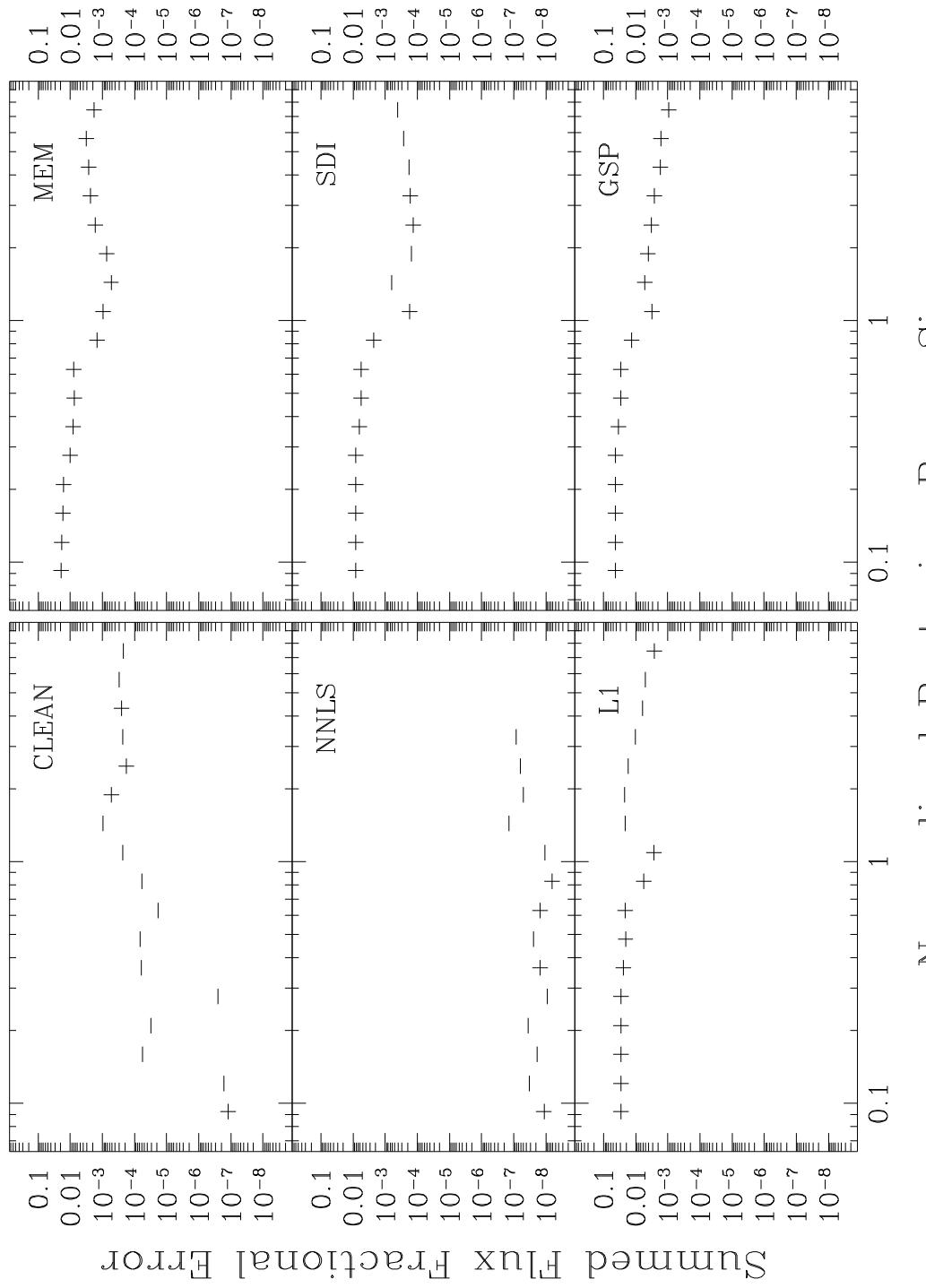


Figure 5.15d: Flux Estimation Errors, Disk Source, Snapshot Coverage

5.7 Gaussian Pixel Functions

I now present a simulation series using the generalized pixel function introduced in Section 2.3. This section is not as thoroughly explored as it might be. In particular, while the generalized pixel function discussed in Chapter 2 is a feature of the discretization of the fundamental convolution equation, and not of any particular algorithm, it is here applied only to CLEAN, and only with a Gaussian pixel function. This technique has been spot checked on NNLS and it seems to help on the extended sources, but it's not yet known how much this practice hurts the algorithmic performance on compact emission.

The most interesting thing about these plots is how much the curves for the different pixel functions resemble the curves for the different algorithms in the algorithmic cross comparison earlier. Compare Figure 5.16 with Figure 5.9. The resemblance is really quite striking! This leads one to speculate that much of the behavior difference seen between algorithms on these resolution test plots is in exactly how much interpixel correlation they implicitly impose.

The one other rather depressing fact about this series is how much the different curves cross each other. The consequence of this being that there is no unique answer to “what is the right pixel function width to use?” if there happens to be emission of many scales in the image. This is hardly surprising, but unfortunate. Since the fitted beam width for this PSF is quite close to $.20''$, notice that even half a beam width in the pixel model completely destroys CLEAN’s ability to deconvolve point sources well. This is probably due to the extended pixel model requiring many more nonzero components to model a point source. The final answer may be reasonable, but we have quite destabilized CLEAN’s convergence. On the plus side, however, a similar small but extended pixel function considerably improves CLEAN’s ability to reconstruct extended emission. It remains to be seen if this is a useful tradeoff on actual sources.

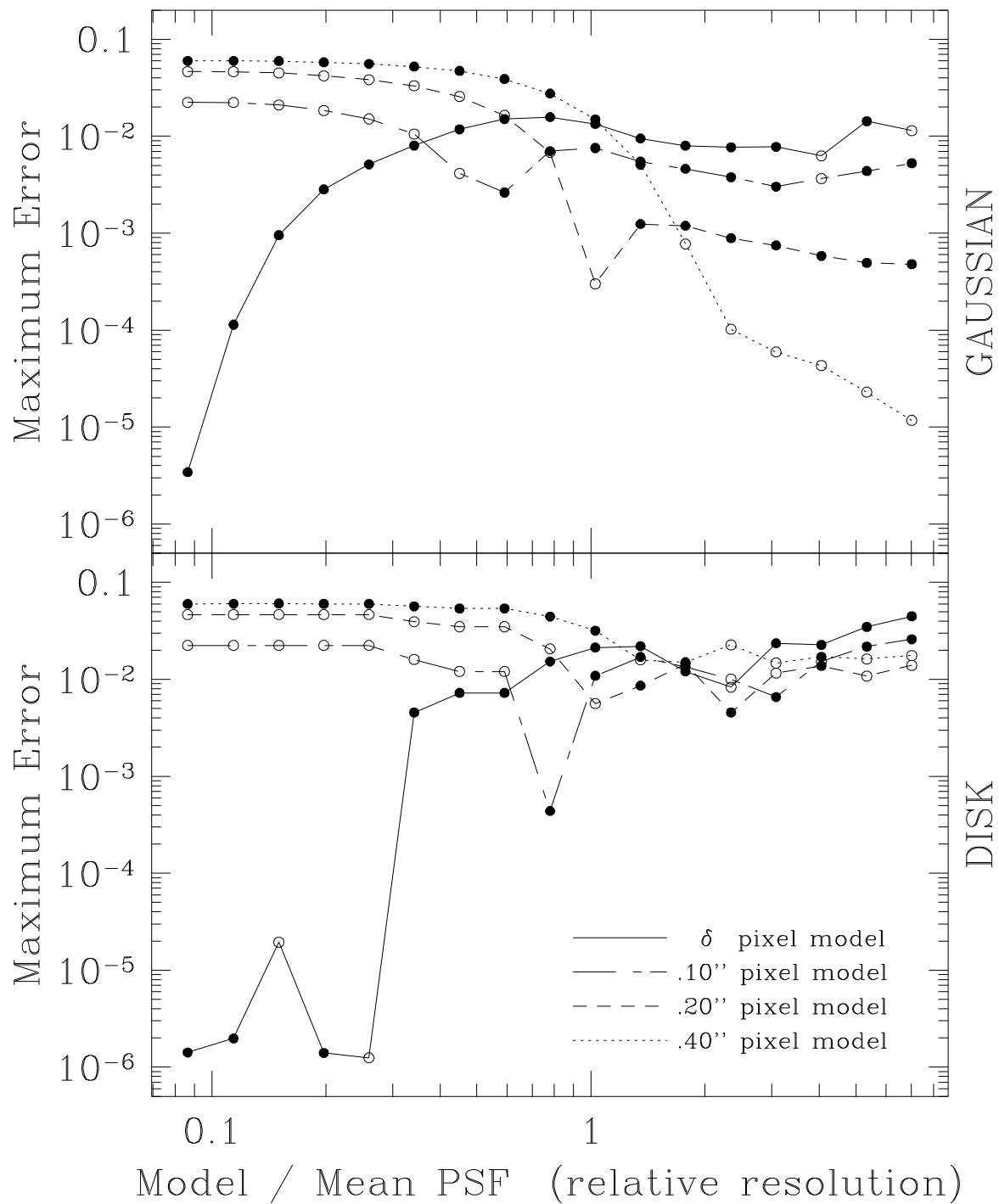


Figure 5.16a: Smooth Pixel CLEAN deconvolution errors against source size.

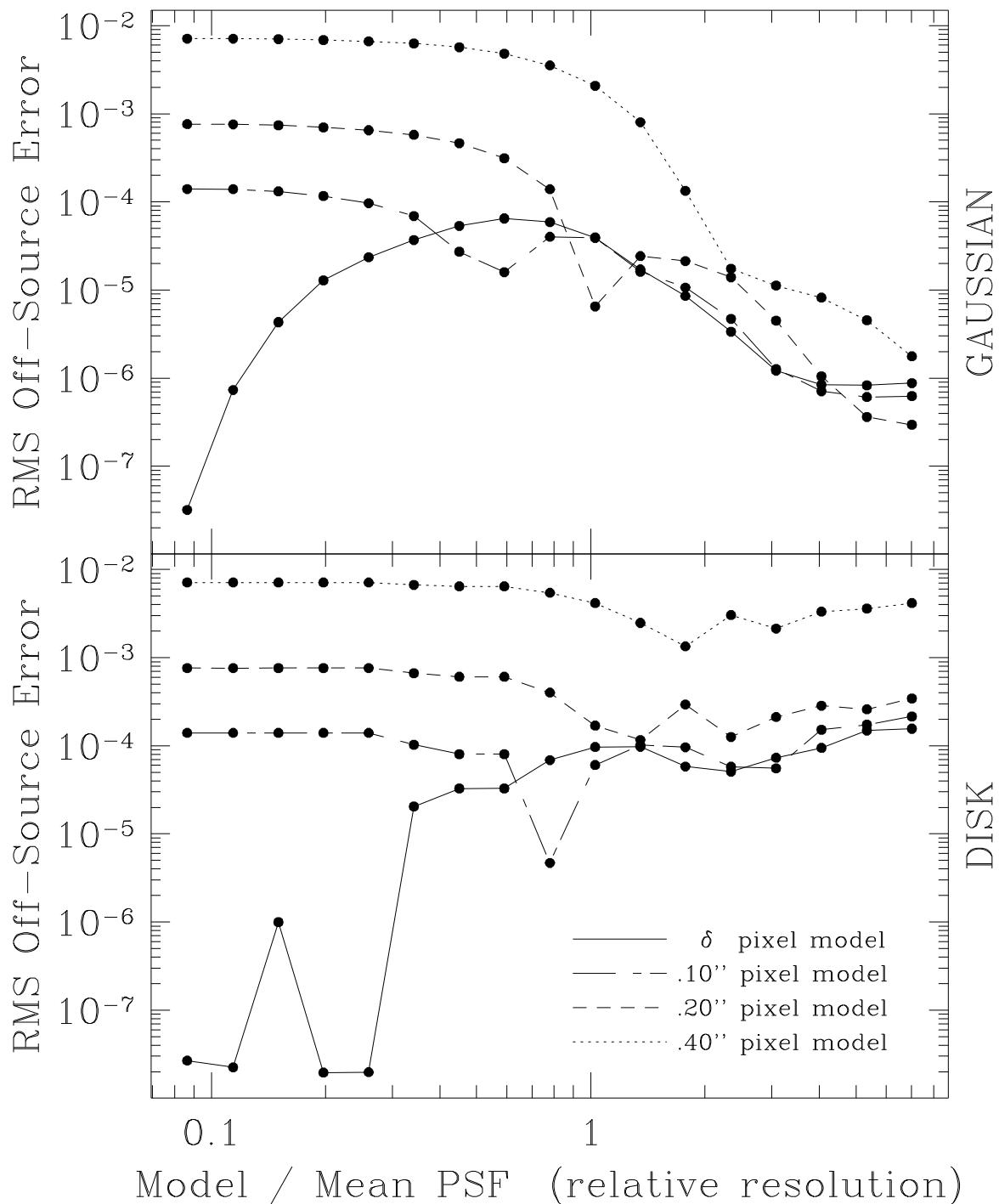


Figure 5.16b: Same as Figure 5.16a, but the off-source root mean square error.

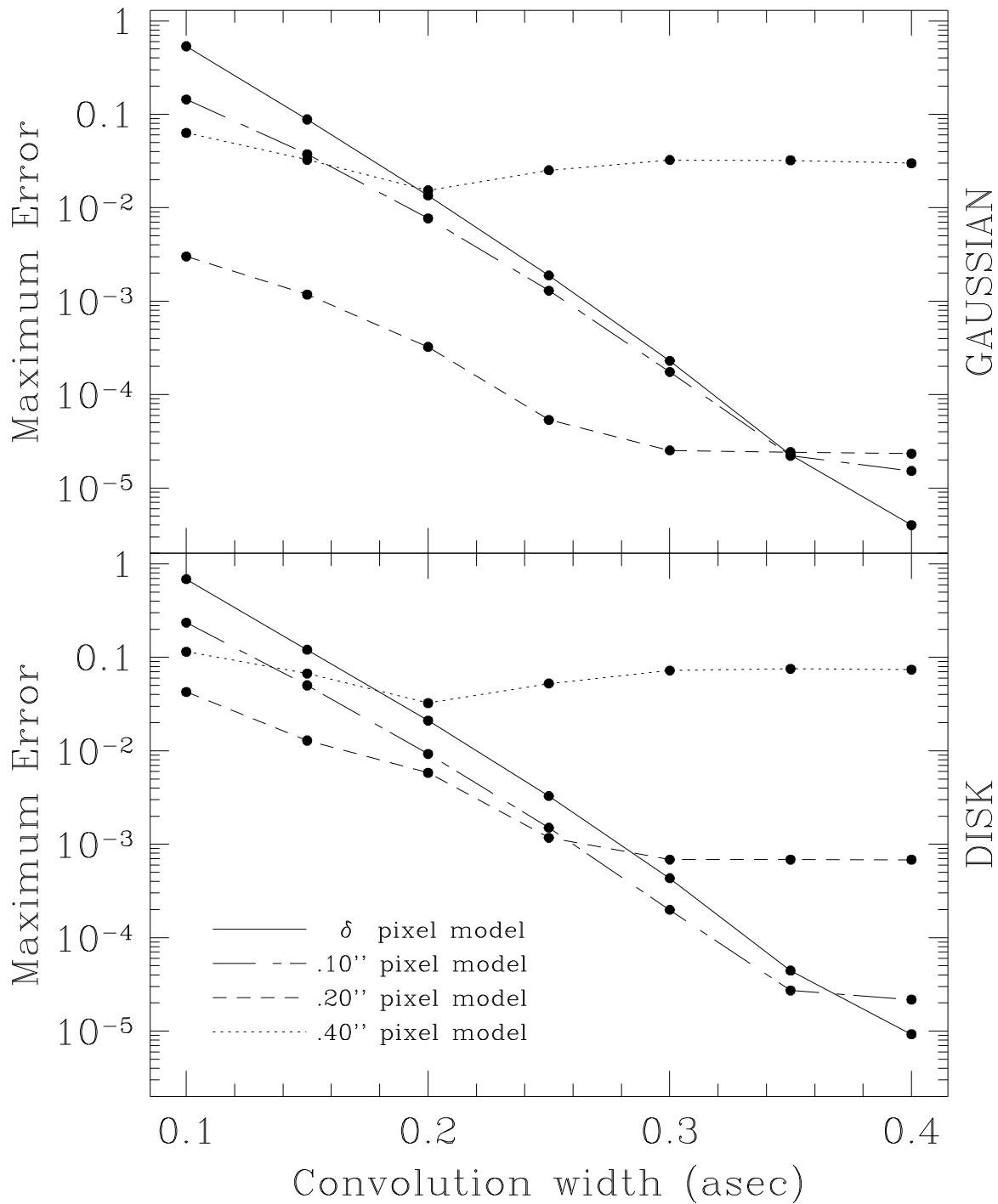


Figure 5.17a: Smooth Pixel CLEAN deconvolution errors against reconvolution size

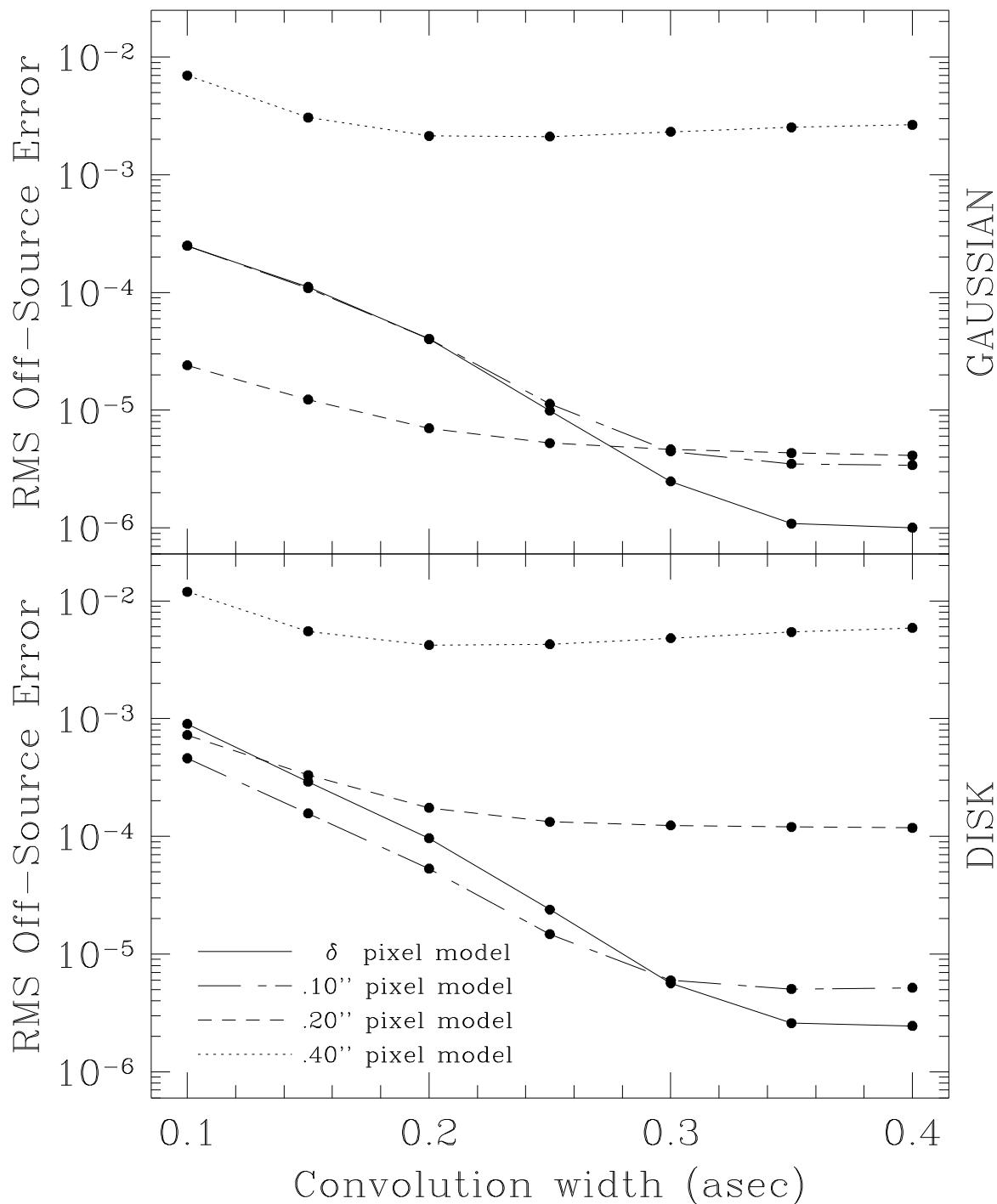


Figure 5.17b: Same as Figure 5.17a, but the off-source root mean square error.

5.8 Point Sources Between Pixels

Next are a few plots which show how well one might hope to do with a point source that has happened to fall between image pixels.

These simulations were done with a data set that resembles the VLA Full Track case, but is slightly different. This is a synthetic VLA observation in ‘C’ configuration. The data was noiseless, free of calibration error, and complete over the full $u-v$ tracks. The simulation frequency was 1.5 GHz, and the declination of the source was 60° . Uniform weighting was used and gave a nearly circular synthesized beam with FWHM of $11.8''$. The coarsely sampled maps used here had a pixel size of $5''$, and the more finely sampled maps used a pixel size of $2.5''$. The dirty image sizes were the usual 512^2 . The $u-v$ coverage and PSF for this observation is given in Figure 5.18. The box surrounding the coverage is the maximum $u-v$ spacing corresponding to a cellsize of $5''$. Clearly this cellsize satisfies the sampling theorem, but not by much. It is a coarser sampling than I would recommend anyone use in practice. The maximum $u-v$ spacing for a cellsize of $2.5''$ is the limits of the plot. This would be quite reasonable for most projects, though somewhat oversampled if the image size were a pressing concern.

The first figure shows how well one could do if the deconvolution algorithms were functioning perfectly. In the centered transform case, the central pixel has merely been set to unity, and the resulting model smoothed to the nominal fitted restoring beam of $11.77''$. The ringing around the source in the coarsely sampled case is merely a reflection of the edge of the gridded $u-v$ plane. The finely sampled case is clearly not having a problem with this effect. The ideal shifted case was generated by simply calculating the analytic transform of a point source shifted by half of one cell in Right Ascension, filling the gridded Fourier plane with this function, back transforming and smoothing as usual. Notice that the ringing around the source is now more severe for the coarsely sampled case, while the smoothly sampled case is very little changed. The analytically expected peak for these two cases is .882 and .969 Jy/beam respectively, so again we are primarily seeing the effects of the reconvolution and of not sampling the function at their peak. If one fits a Gaussian to the finely sampled source, the total flux recovered is 1.0000 Jy. The coarsely sampled source cannot sensibly be fit — too few pixels are above the fitting threshold of .35 times the peak. Resorting instead to summing the flux out to a radius of 10 pixels in the coarse case and 20 pixels in the fine, both cases recover 1.000 Jy. The total flux in the ringing at the source edge cancels out to a high degree of accuracy.

Things get interesting when we involve the deconvolution algorithms, as shown in Figures 5.19b and 5.19c. The greyscale is two sided logarithmic around 0, and is common to all three panels. CLEAN was run for 10000 iteration at a loop gain of 0.1, with no support constraint, for a maximum residual of 81 and $22 \mu\text{Jy}/\text{beam}$ for the coarse and fine cases. MEM was run for 100 iterations, with `Flux = -.1` and `Sigma = 10^{-5}` , yielding maximum residuals of 46 and $19 \text{ mJy}/\text{beam}$. NNLS was run with data and windows of 40 pixel radius and produced residuals of and 62 and $19 \text{ mJy}/\text{beam}$. Clearly, both MEM and NNLS are

having difficulties with the negatives in the components that are required to fit the data. In this case, the positivity constraint in these algorithms is a distinct disadvantage.

Summing the image pixels out to the radii mentioned, CLEAN gives a summed flux of 1.000 and 1.000 Jy for the coarse and fine images. Fitting to the image, we are sampling the shape of the restored peak more than the ringing around the base. CLEAN is not quite as good here, and the flux in the fitted Gaussian is 1.0044 Jy. With MEM, the sum for the coarse image is 1.229 Jy! The sum for the finely sampled image is somewhat better, at 1.074 Jy, but still really quite poor given that this is noiseless data. Interestingly, the fitted flux for MEM is only 1.0283 Jy – it does a better job of reconstructing the peak than it does near the base. This isn't really surprising, since that's where the positivity constraints are best being satisfied. NNLS is intermediate between these extremes. The summed flux is 1.150 and 1.025 Jy for the coarse and fine images, and the fitted flux is 1.0160 Jy.

Positivity constraints are only a benefit when the image to be restored obeys them! Due to representation effects, it may not. CLEAN is the best choice when the image requires significant negatives.

Finally, in Figure 5.19d we can see the model that the algorithms were trying to recover in the finely sampled case. With negatives 30% of the peak, it's no great wonder that the positivity constrained algorithms were having trouble. Notice that MEM and NNLS have taken the one-dimensional ringing pattern and expanded the components out into two dimensions while trying to fit the negatives.

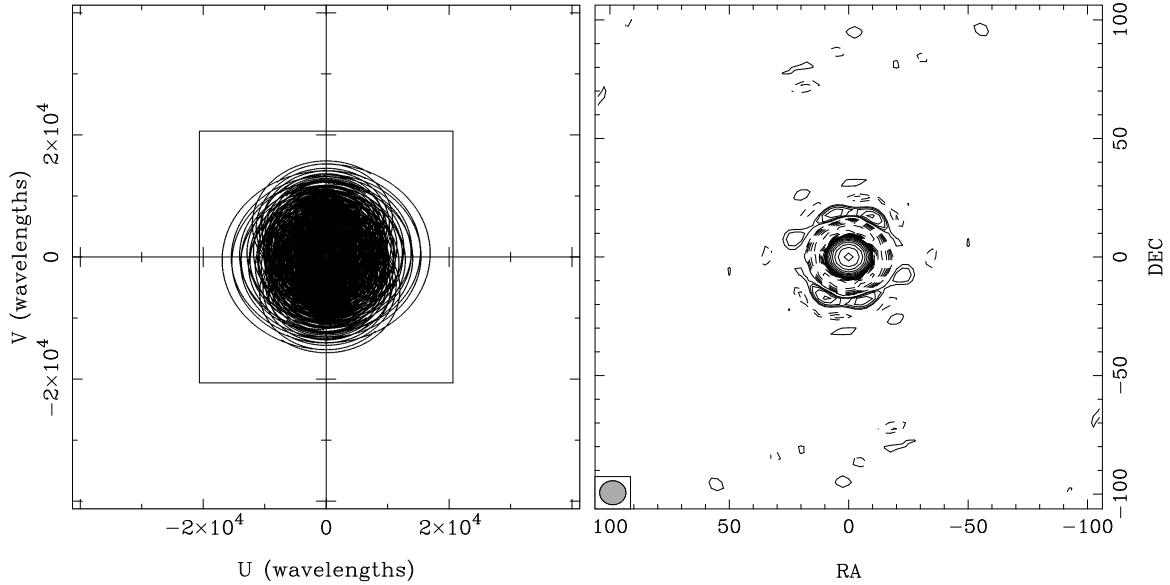


Figure 5.18: Shifted Point Source — u - v coverage and PSF

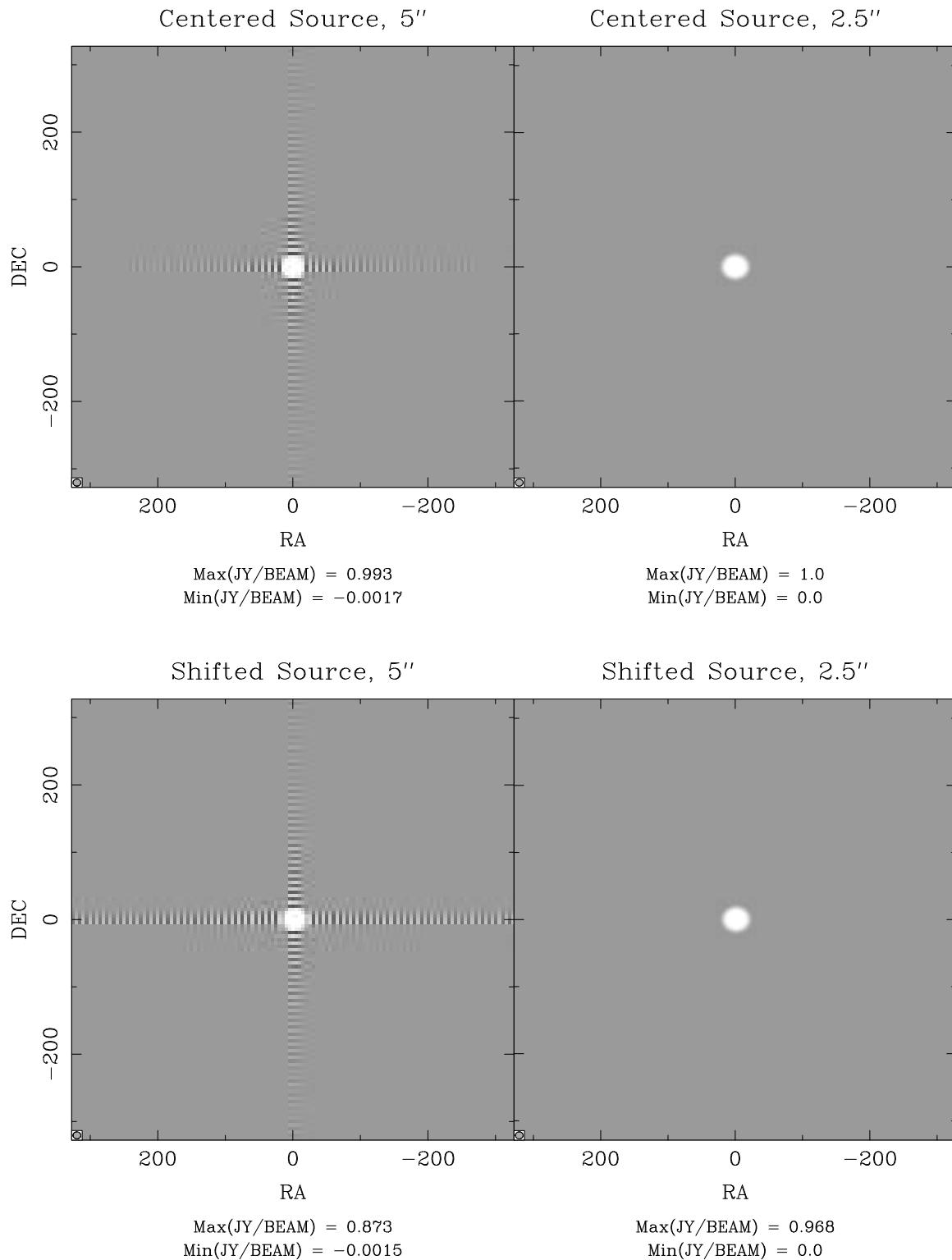


Figure 5.19: Limits set by the finite cell size. The model is perfect and the artifacts are generated by the convolution with the restoring beam. Greyscale is two sided logarithmic around zero and common to all panels.

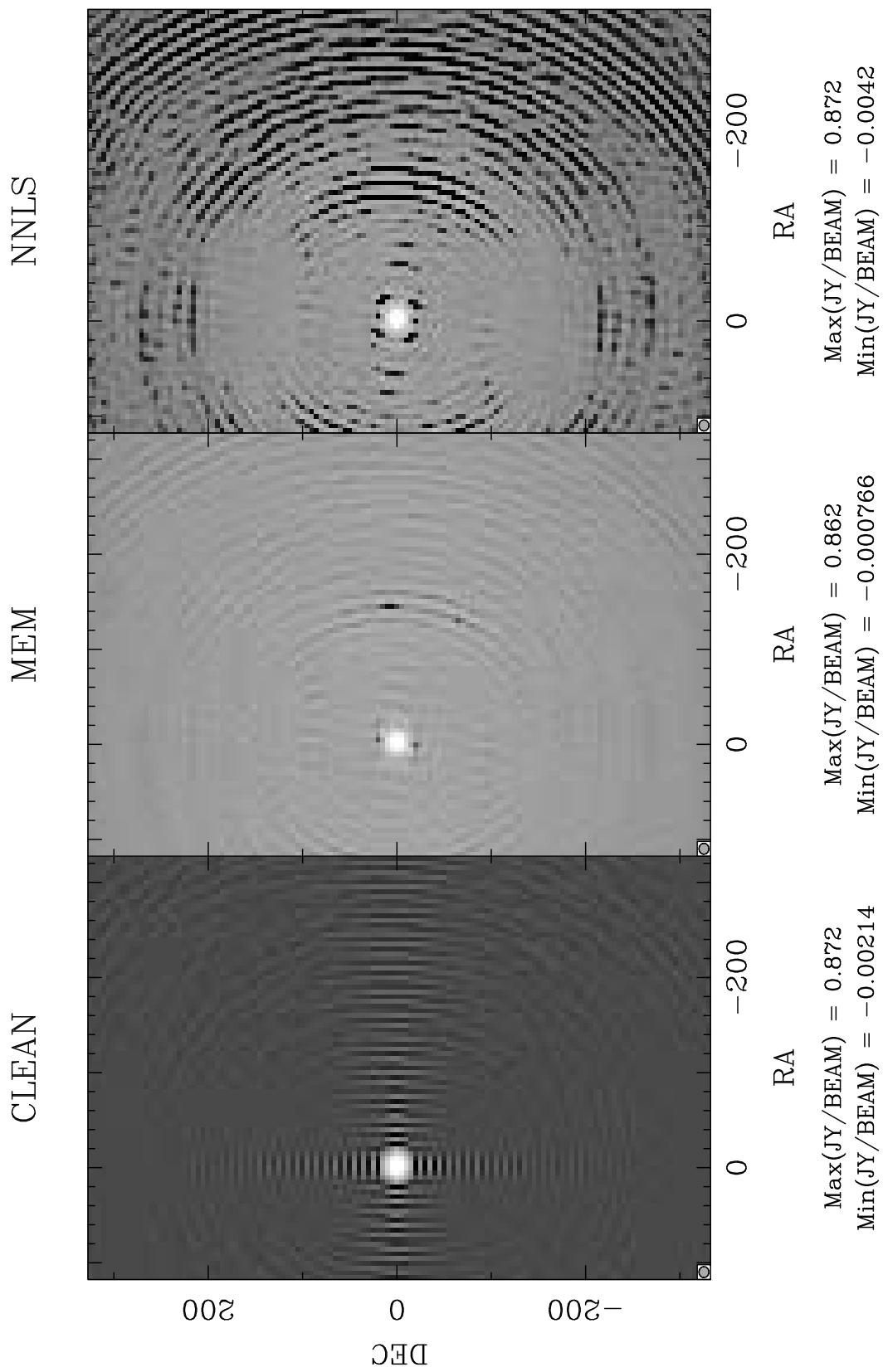


Figure 5.19b: Deconvolved images of a shifted point source, coarse sampling.

NNLS

MEM

CLEAN

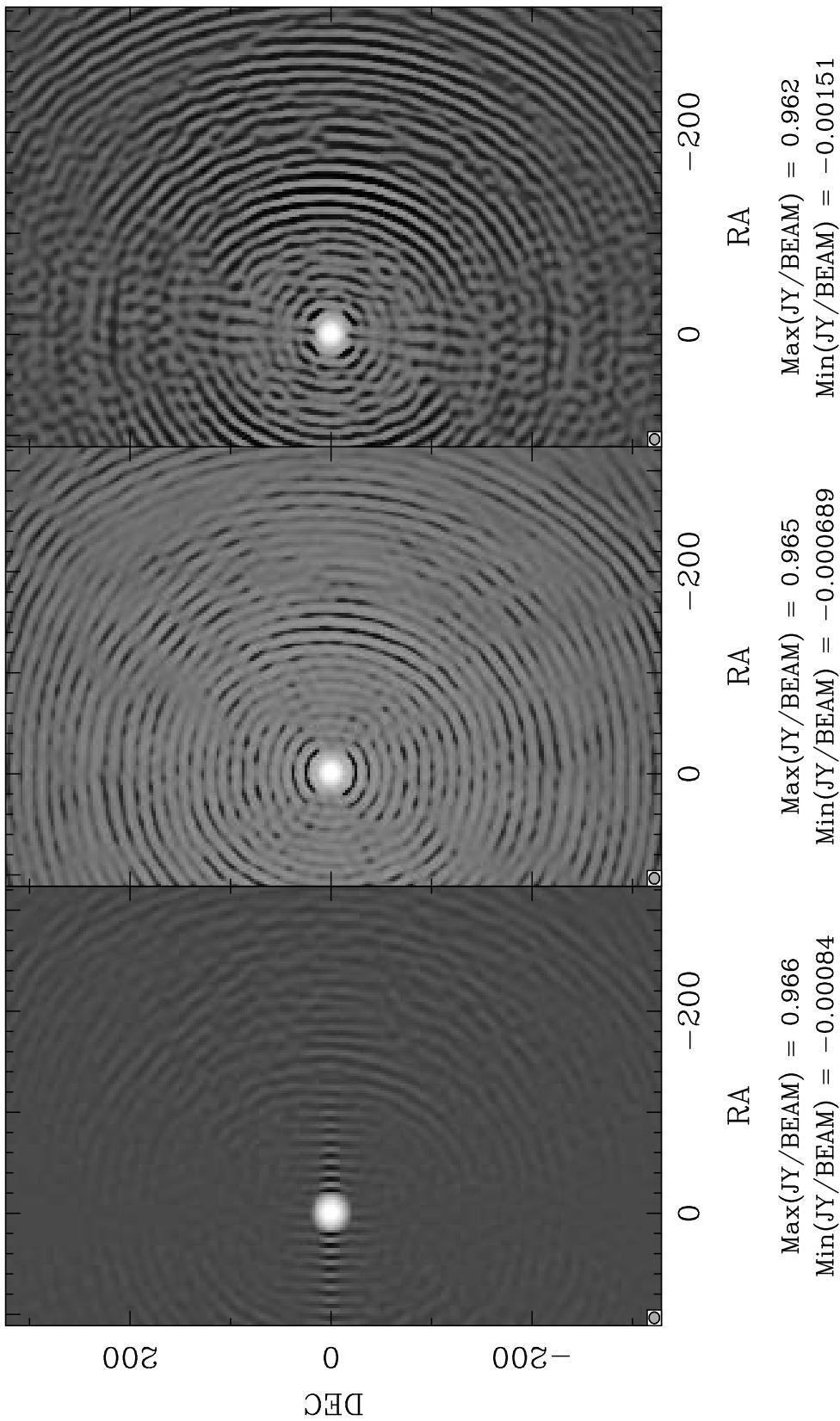


Figure 5.19c: Deconvolved images of a shifted point source, fine sampling.

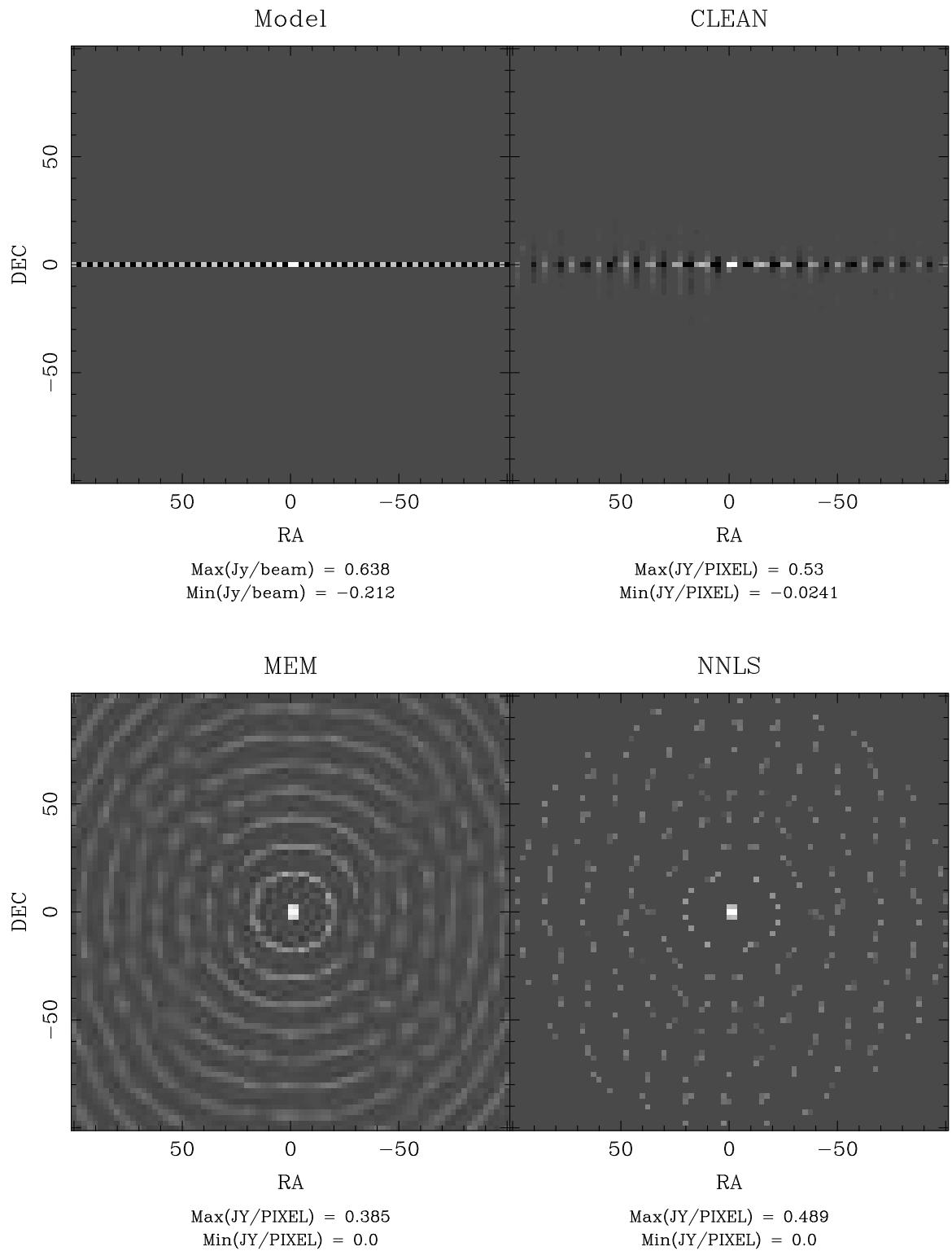


Figure 5.19d: Component models of a shifted point source, fine sampling.

5.9 Convergence in the Fourier Plane

We now make a brief diversion to examine convergence rates as a function of position in the u - v plane. This is for the same shifted point source simulations as in Section 5.8. Figures 5.20a and 5.20b show multiple traces, each corresponding to a particular point in the u - v plane. Since this is a unit point source model, the true model amplitude is 1.000 at all spacings. The particular points chose were all along the $v = 0$ axis, and have u values of $0 \text{ k}\lambda$, $2 \text{ k}\lambda$, $4 \text{ k}\lambda$, and so forth out to $20 \text{ k}\lambda$. The maximum value of u in the data set is $17 \text{ k}\lambda$, so the last two traces are in the extrapolated region. For each of the algorithms, the traces are given for the $2.5''$ cellsize and the $5.0''$ cellsize. Additionally, each is given for a Gaussian pixel function which is fitted to the size of the beam.

Granted, these are rather extreme cases, involving extreme critical sampling, an extreme number of components in the true model that the algorithm is trying to recover, and extremely large pixel functions in the very case where the algorithm is trying to recover high frequency structure. Few real cases are likely to be this bad. But the interesting thing is how long CLEAN takes to converge at the outer u - v spacings. Clearly, the inner spacings converge more quickly than outer spacing, perhaps not a great surprise. But few people would have predicted that it would require 10^4 iterations for CLEAN to converge on the longest spacings for a single point source. Cutting the cell size down has helped speed the process, so again — when faced with representation problems, oversample. The Gaussian pixel plots show that the smooth pixel function has considerably destabilized the convergence process. The final solution may in fact be better than the delta-function pixel functions, and almost certainly will be on smooth emission, but getting there may well take much longer. The MEM plots are harder to interpret. I find the bifurcation in the delta-function, coarsely sampled case to be particularly interesting, but I do not know what is causing it.

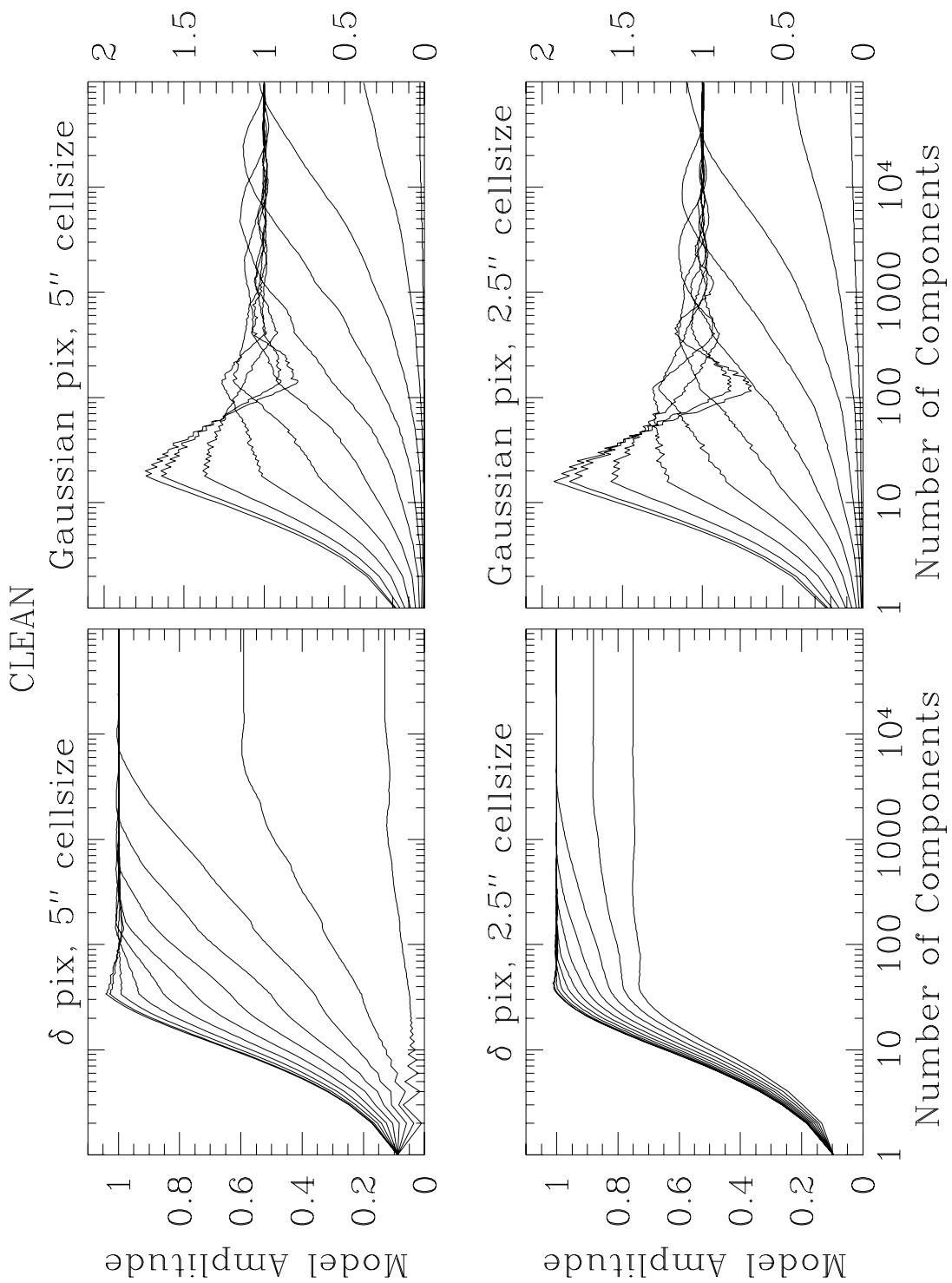


Figure 5.20a: Shifted point source — u - v plane convergence for CLEAN. The true model amplitude is unity. Reading from the top, the traces correspond to $(u, v) = (0, 0), (2k\lambda, 0), (2k\lambda, 0), \dots$

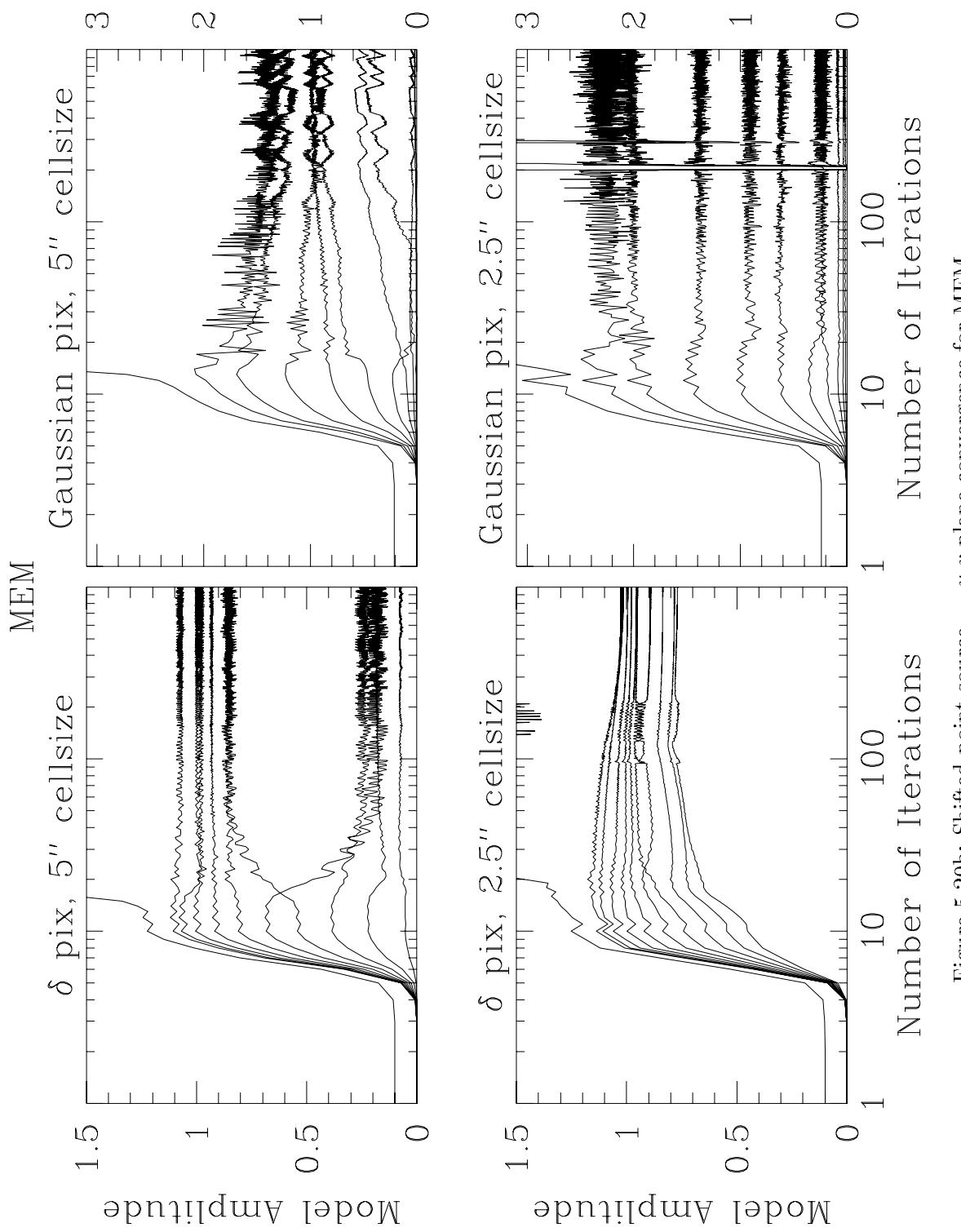


Figure 5.20b: Shifted point source — u - v plane convergence for MEM

5.10 3C48 Simulations — MEM

This dissertation has been concentrating on CLEAN, MEM and NNLS as the three deconvolvers most likely to be used in radio interferometry. The 3C48 test case has already been presented for CLEAN in Section 3.10, and the similar plots for NNLS in Section 4.3.3. Here we present the corresponding MEM plots for completeness. As before, model parameters and plots are found in Table 3.8, and Figure 3.30b. The noisy cases were run for 100 iterations with $T_{\text{flux}} = -0.001$ and $\text{Sigma} = 3(10)^{-3}$. These have certainly converged completely. The noiseless cases were run for 1000 iterations each, with the same T_{flux} and $\text{Sigma} = (10)^{-6}$. Conceivably these might benefit from further iteration, though the solution was improving only very slowly with iteration. The asymptotic limit is probably not terribly different from the numbers shown here.

Most of the salient points have been made in the previous presentations, but for MEM specifically:

- MEM has a real problem with naturally weighted beams. It's most obvious in the noiseless cases, but even in the on-source noisy case, the naturally weighted beam has the worst error.
- MEM does adequately with the uniform beam. The uniform beam produces the smallest off-source RMS error in the noiseless case, and an intermediate value of full-image maximum error. When noise is added, the maximum error for the uniform beam approaches that for the natural beam.
- MEM does fairly well with the robustly weighted beams. The best-fit-Gaussian beam, ($\text{Robust} \approx 0.6$, $\text{RMS} = 1.09$), has the minimum error in some cases, and not terribly far from it in the rest. CLEAN seems to be the only major algorithm which occasionally has problems with a robust beam in the high dynamic range regime.
- MEM is simply not very good at removing the far sidelobes of compact sources. In the noiseless case, it is a full factor of 10 worse than CLEAN and nearly three orders of magnitude worse than NNLS. This is still good enough to be masked by the thermal noise in the noisy case.

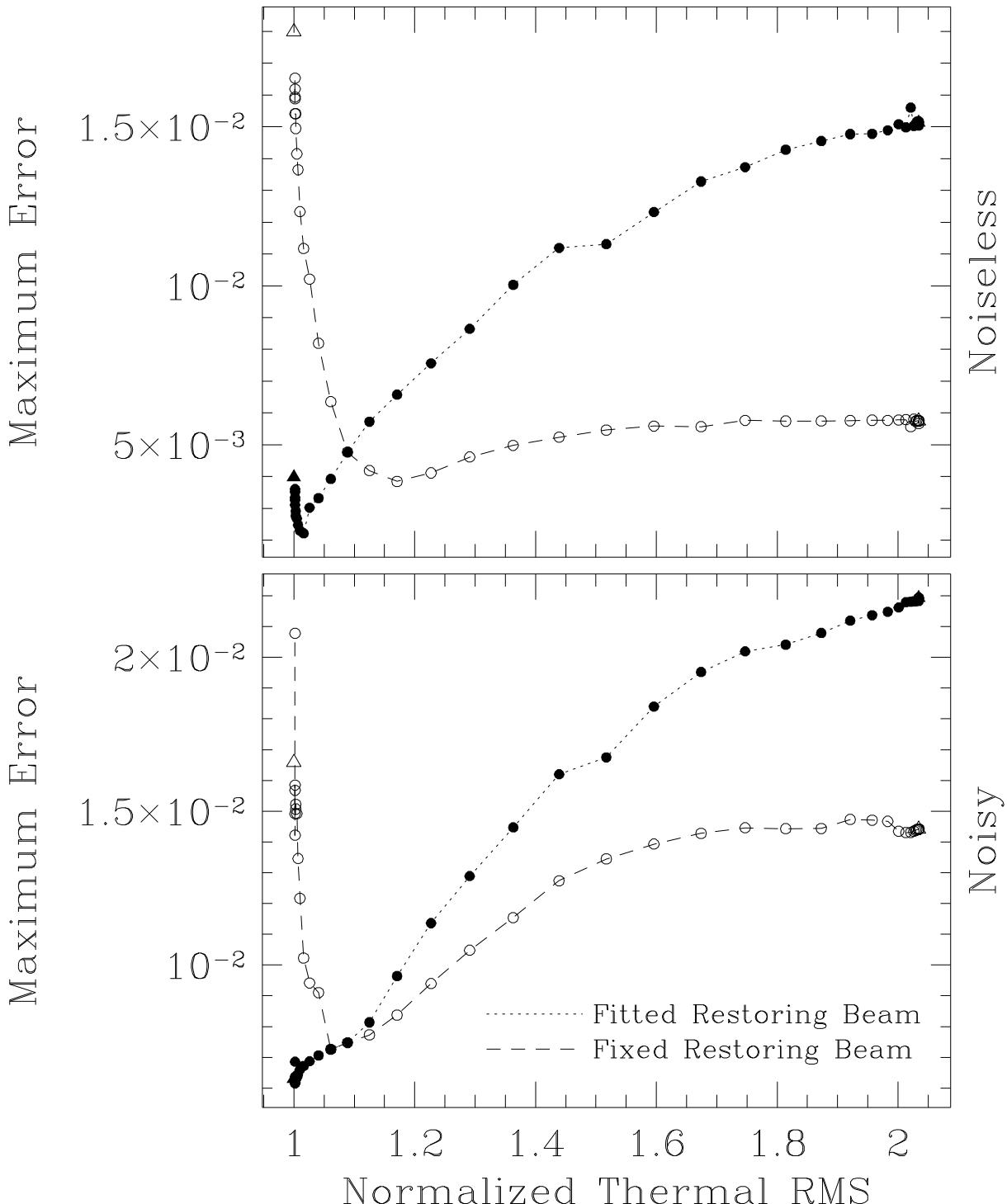


Figure 5.21a: Error quantities against RMS. The fitted restoring beam corresponds to Figure 3.30b. The fixed beam curve is more representative of the absolute deconvolution quality and used $\sqrt{B_{nat}B_{uni}} = .2214''$.

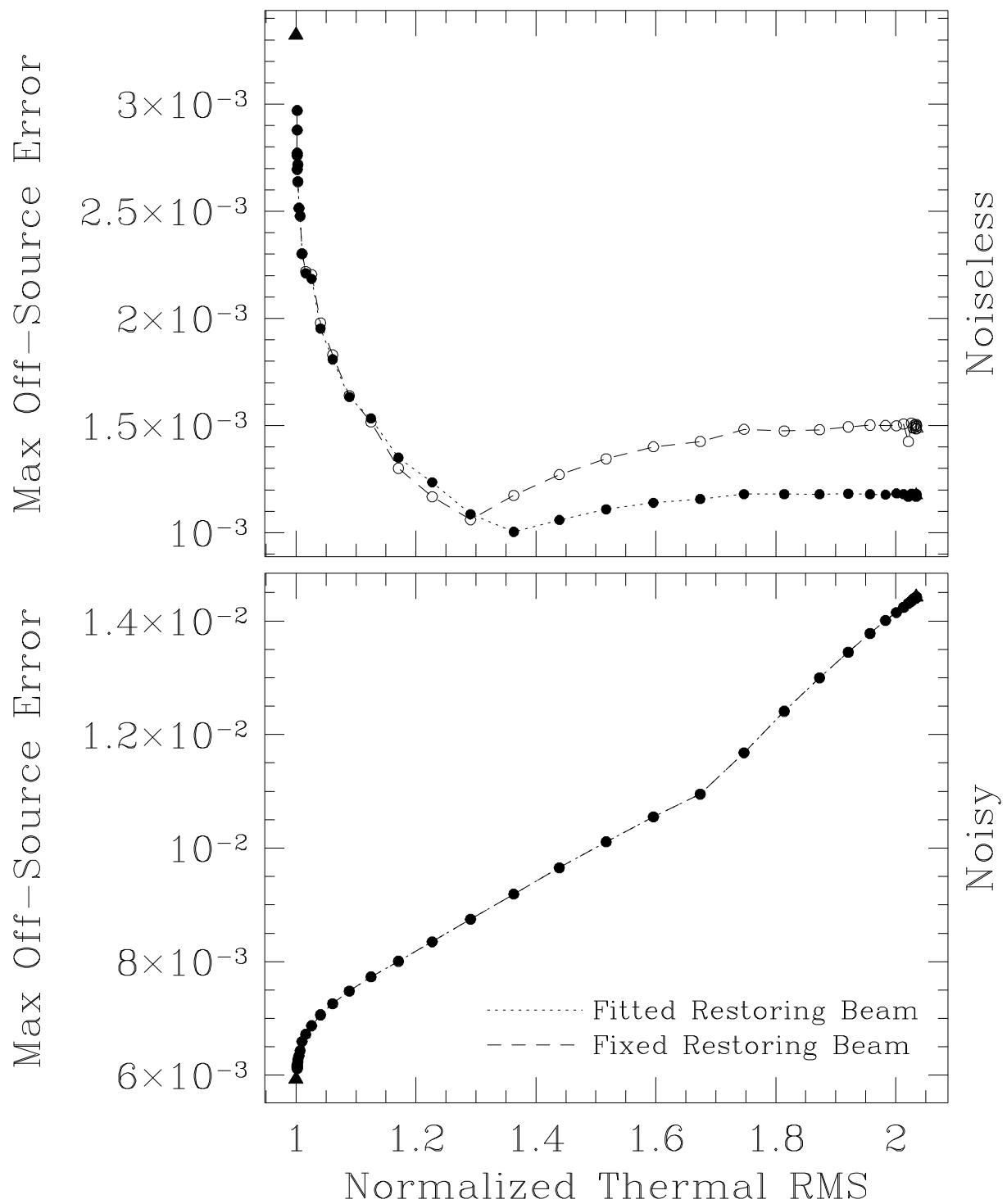


Figure 5.21b: Error quantities against RMS degradation factor: maximum off-source error.

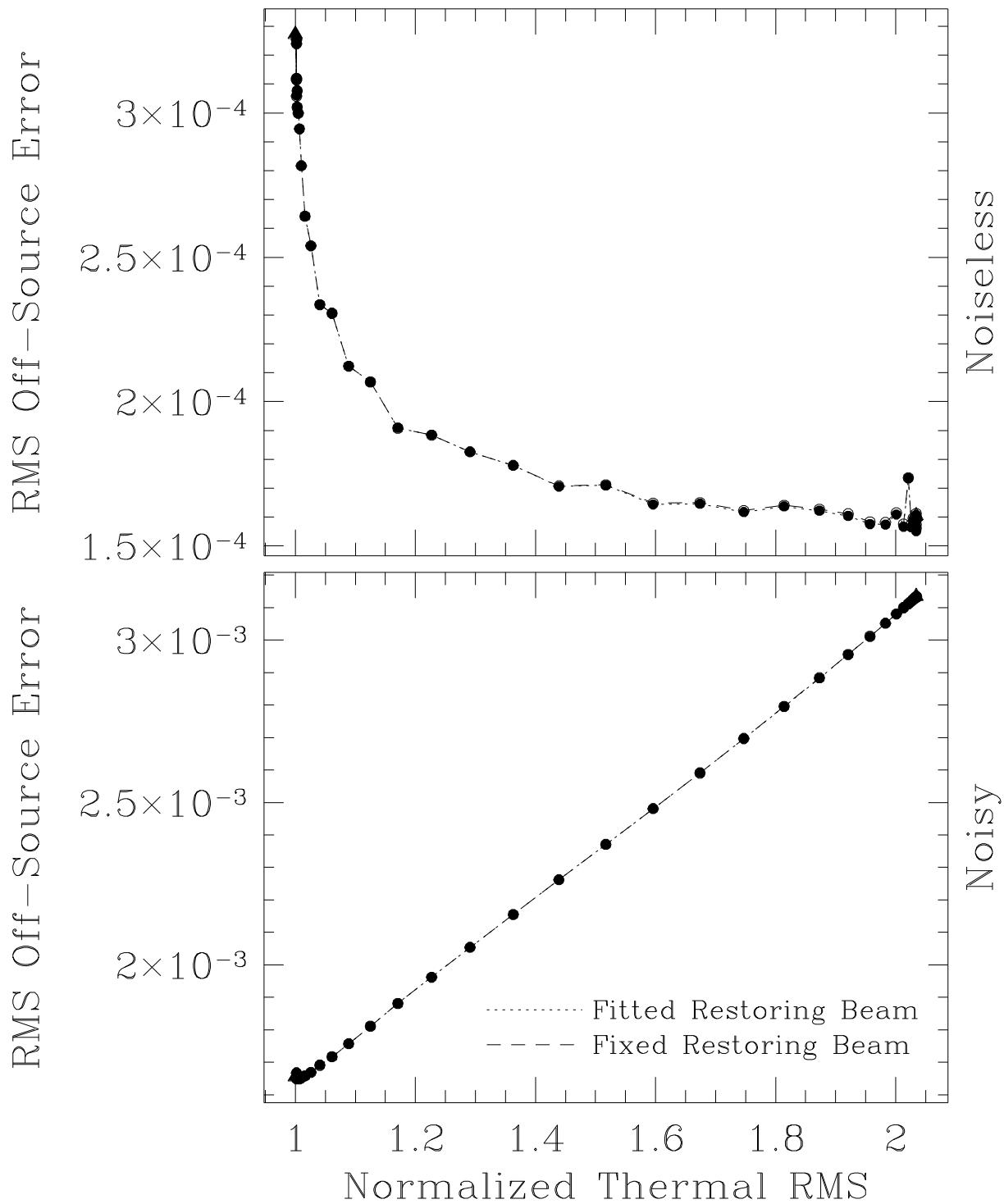


Figure 5.21c: Error quantities against RMS degradation factor: off-source RMS error.

Chapter 6

Miscellaneous Topics

This chapter is a small catch-all both of topics unrelated to those covered in previous chapters, and of several “loose ends” — material that was not judged important enough to interrupt the flow of more germane material earlier on. Specifically, we have a method of estimating noise from visibility data, some semi-analytic results about how CLEAN places its components around compact emission, a result which directly establishes the connection between CLEAN components and the Fourier coefficients of the visibility data in a particular limiting case, and a discussion of combining models from different deconvolution algorithms.

6.1 Estimation of Thermal Noise from the Visibility Data

Back in Chapter 3, much use was made of the proportionality constant between weights and inverse thermal noise variance. This quantity was needed to predict the image plane thermal noise, and hence to gauge the quality of the deconvolution/calibration by how closely this is achieved. From before, the constant ΔS_{uw} is the expected RMS thermal noise on a single correlator channel corresponding to any visibility of unit weight. $w_k = \Delta S_{uw}^2 / \sigma_k^2$, where σ_k^2 is the variance of the thermal noise in the real or imaginary part of a visibility with weight w_k . σ_k is often called ΔS in the usual references, and can be calculated a priori via the known antenna calibration and the observation parameters. The problem is that such a priori calculation of ΔS_{uw} is rarely known accurately enough. With careful a priori calibration of the instrument, one can do well enough to diagnose many gross errors. But when wondering about differences of tens of percent between predicted and measured thermal noise, a priori sensitivity calculations are rarely sufficient.

One common expedient is to use the measured RMS in the Stokes V dirty image. Since in nearly all cases the Stokes V image is expected to be source free, the dirty image RMS is exactly the desired image plane thermal RMS. When the polarization data are available, this is a perfectly reasonable procedure. But occasionally it is not, or one may simply wish to avoid the complications of processing multiple Stokes parameters. There is another alternative that is equally simple, which estimates the thermal noise from the internal scatter in the visibility data. One can time average the visibility data, estimate the thermal noise in the mean sample by the sample variance and then accumulate $w_k \sigma_k^2$ over all the averaged samples as estimates of ΔS_{uw}^2 . This works well, though one must worry

about the proper time bin size and hope that the changing source structure does not bias the measured variance. Better still is to use a successive difference approach, which is really just a degenerate case of time averaging sample variances. The important thing to note is that if X_1 and X_2 are the real or imaginary part of two visibility measurements with common mean, then $(X_1 - X_2)/\sqrt{1/w_1 + 1/w_2}$ is a Gaussian random variable with mean zero and variance ΔS_{uw}^2 . This can be trivially accumulated over the entire visibility set, summing over pairs of adjacent points on the same baseline and skipping pairs separated by too large a time difference. Clearly adjacent pairs of differences are correlated, but it is easy to show that the expectation value of the sample variance with zero mean is ΔS_{uw}^2 .

$$\begin{aligned}
 \left\langle \frac{1}{N} \sum_{i=1}^N D_i^2 \right\rangle &= \frac{1}{N} \sum \left\langle (X_{i+1} - X_i)^2 \right\rangle \\
 &= \frac{1}{N} \sum \frac{\langle X_{i+1}^2 \rangle + \langle X_i^2 \rangle}{1/w_{i+1} + 1/w_i} \\
 &= \frac{1}{N} \sum \frac{\sigma_{i+1}^2 + \sigma_i^2}{1/w_{i+1} + 1/w_i} \\
 &= \frac{1}{N} \sum \left(\frac{w_{i+1}\sigma_{i+1}^2}{1+w_i/w_{i+1}} + \frac{w_i\sigma_i^2}{1+w_{i+1}/w_i} \right) \\
 &= \frac{\Delta S_{uw}^2}{N} \sum \left(\frac{1}{1+w_i/w_{i+1}} + \frac{1}{1+w_{i+1}/w_i} \right) \\
 &= \frac{\Delta S_{uw}^2}{N} \sum (1) \\
 &= \Delta S_{uw}^2
 \end{aligned}$$

One can get an idea of how stationary the source visibility must be to avoid corrupting the measured noise by considering just one difference. If

$$\langle X_1 - X_2 \rangle = \mu_1 - \mu_2 = \varepsilon,$$

then it is simple to show that

$$\langle (X_1 - X_2)^2 \rangle = \sigma_1^2 + \sigma_2^2 + \varepsilon^2$$

Thus if $\sigma_1 = \sigma_2 = \sigma$ and we wish to bias the measured standard deviation by less than a factor of $f \ll 1$, we require

$$\left(\sqrt{\sigma^2 + \varepsilon^2/2} - \sigma \right) / \sigma < f \quad \Rightarrow \quad \varepsilon/\sigma < 2\sqrt{f}$$

This is a very favorable limit. If the bias must be less than 10%, we can tolerate the visibility real or imaginary parts varying between adjacent measurements by as much as 63% of the single sample RMS noise. There will be almost no contamination from source structure near the phase center in real problems. Flux far from the phase center may need to be subtracted from the visibility data set prior to noise estimation, however.

Successive differences is generally the method of choice, but averaging over finite time bins does have one virtue. One can use a median or more exotic average rather than a weighted mean to estimate the averaged visibilities. If the input noise is Gaussian, the median average will throw away more signal than is necessary, and will degrade the measured noise in a known way.¹ However, if the input noise is *not* Gaussian, then the median averaged data will degrade less than expected or even improve. The idea is that confusing sources far from the phase center will vary much more quickly than the program source due to phase wind, and will effectively constitute a noise source rather than a stationary signal over the averaging time. Other sources such as terrestrial interference will likewise have a very nonstationary, non-Gaussian structure.

A possible procedure starts by estimating the thermal noise with successive differences. Average the data down in time, using both a mean and median averaging procedure. Estimate the noise again from the averaged data using either successive differences or internal scatter during the average. Is the ratio of the estimated noise in the median averaged set to the mean averaged set less than expected for Gaussian data? If so, the visibility data itself is corrupted by nonthermal effects. If one chooses to measure the noise in the image plane after time averaging, the measured RMS will be potentially contaminated by both non-Gaussian noise in the visibility data itself, and by image plane effects like sidelobes due to calibration errors and simple deconvolution errors. In principle, both the image plane and visibility based effects could be separated by the different measurements of the noise.

Finally, if the input statistics are assumed to be Gaussian, we can easily calculate the variance in the sample variance itself. This yields a lower limit on the standard error of ΔS_{uw} calculated from successive differences, namely

$$\sigma_{\Delta S_{uw}} = \Delta S_{uw} / \sqrt{2N}$$

Here N is the number of samples used to estimate ΔS_{uw} , counting real and imaginary correlator channels separately. This is a lower limit, and if the noise statistics are not Gaussian the standard error will be higher. If two estimates from different subsets of the data agree to within this error, one can consider the data subsets equivalent in terms of non-Gaussian noise contamination. If the estimates do not agree to this tolerance, the data subsets may be affected differently by source structure, the noise may not be Gaussian, or both.

¹Precise values are found in Kendall & Stuart (v. 2, p. 7, 1977). The degradation of the variance in the limit of large sample size is $2/\pi$. Unfortunately, all the formulae referenced assume that the samples are drawn from the same Gaussian parent population, i.e. that they have the same weights. I know of no analytic estimate of the variance degradation in the unequal weights case. Fortunately this is rarely a problem in practice, since the points along a given baseline will nearly always have the same weights. If this does become a problem, one could use the bootstrapped variance as described by Efron and Tibshirani, (1991).

The mean and median boxcar time averages and successive difference noise estimation are all available in the SDE task, `visavg`. The full possibilities of these techniques have not been investigated particularly well, however. Preliminary results from the case studies in chapters 7–9 are that thermal noise can be estimated to within a few percent. At this level there are often systematic variations with baseline length larger than accountable by Gaussian noise statistics. Even this is better than typically available *a priori*, so the technique seems quite promising.

6.2 CLEAN component placement

Since CLEAN is so prevalent in radio astronomy, it has been investigated somewhat more carefully than the other deconvolution algorithms in this dissertation. I present here several modest analytic results that clarify some of the observed behavior, and also a detailed simulation of the consequences of this behavior on the fitted flux to isolated resolved sources.

While some progress has been made with the optimization criteria implicitly satisfied by CLEAN, (Marsh and Richardson, 1987), there is relatively little work on direct understanding of CLEAN component placement. Previous work by Schwarz (1984) concentrated on the well known problem of ‘CLEAN stripes’, where regularly spaced corrugations appear superimposed over smooth emission. The power corresponding to these corrugations appears in unsampled regions of the u - v plane. That is, the stripes are not constrained by the data. The algorithm must estimate the correlated flux at the unmeasured spacings, and is not doing a particularly good job. The explanation given by Schwarz is that after the first subtraction of the beam from smooth emission, the negative first sidelobes of the dirty beam gives rise to local maxima at the position of those sidelobes. Later iterations will place components at these induced maxima and the process repeats, leading to periodic placement of the CLEAN components at a spacing of the distance between the first sidelobes. The first iteration of this process is shown in Figure 6.1. Here, an attempt is made to provide a similar qualitative explanation for the opposite limit to component placement on smooth emission — the component placement around compact peaks.

6.2.1 The Gaussian approximation to initial CLEANing

We can gain a qualitative understanding of the initial component placement for semi-compact sources by considering the following simplified problem.

Let the full width at half maximum of a Gaussian be σ , so that

$$G(x) = k \exp(-cx^2/\sigma^2), \quad \text{with } c = 4 \ln 2.$$

Both the source and the dirty beam are assumed to be Gaussians, with half widths σ_S and σ_B , respectively. The dirty map is the convolution of G_S and G_B , which is again a

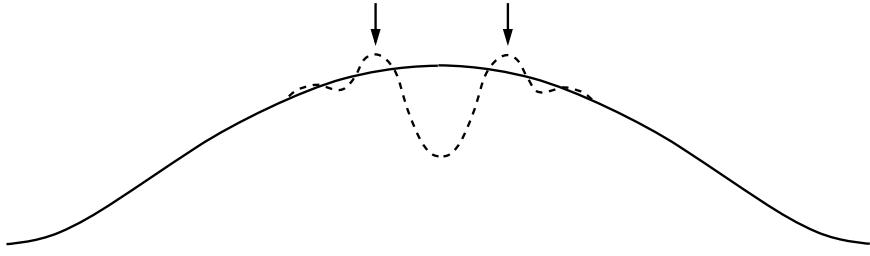


Figure 6.1: Creation of local maxima in smooth emission, adapted from Schwarz (1984). The initial smooth dirty image is the solid line, and the residual after the first iteration of CLEAN is the dotted line. Subsequent CLEAN components will be placed at the locations of the arrows and the regular placement of components will propagate across the emission.

Gaussian of half width $\sigma_D = \sqrt{\sigma_S^2 + \sigma_B^2}$. CLEAN is a shift invariant process, so we place the source at the origin. We ignore the discretization for the moment, and treat the functions as continuous.

Consider a point where the CLEANing process has placed flux only at the origin. Less a leading constant, the residual map will be

$$R(x) = \exp(-cx^2/\sigma_D^2) - \xi \exp(-cx^2/\sigma_B^2),$$

where ξ is the fractional CLEAN depth and will depend on the loop gain and how deeply the CLEAN has progressed. At each iteration, CLEAN will pick the point of maximum residual which is a local maximum. Thus we differentiate

$$\frac{\partial R}{\partial x} = -2xc \left[\frac{1}{\sigma_D^2} \exp(-cx^2/\sigma_D^2) - \frac{\xi}{\sigma_B^2} \exp(-cx^2/\sigma_B^2) \right]$$

$$\frac{\partial R}{\partial x} = 0 \quad \Rightarrow \quad x = 0 \quad \text{or} \quad \frac{1}{\sigma_D^2} \exp(-cx^2/\sigma_D^2) = \frac{\xi}{\sigma_B^2} \exp(-cx^2/\sigma_B^2)$$

Since we are interested in the first minimum off the central peak, we keep only the latter case, and solve for the location of the maximum x_m .

$$\xi \left(\frac{\sigma_D^2}{\sigma_B^2} \right) = \exp \left(cx_m^2 \left(\frac{1}{\sigma_B^2} - \frac{1}{\sigma_D^2} \right) \right)$$

Defining the ratio of source to beam size $\delta_S \equiv \sigma_S/\sigma_B$, this becomes

$$\begin{aligned} \ln(\xi(1 + \delta_S^2)) &= c \left(\frac{x_m}{\sigma_B} \right)^2 \left(1 - \frac{1}{1 + \delta_S^2} \right) \\ \frac{x_m}{\sigma_B} &= \left[\frac{\ln(\xi(1 + \delta_S^2)) (1 + \delta_S^2)}{c} \right]^{1/2} \end{aligned} \tag{6.1}$$

Clearly there is a critical point when $\xi(1 + \delta_S^2) = 1$. Define $\xi_c \equiv 1/(1 + \delta_S^2)$, and note that for $\xi < \xi_c$ there are no external maxima, and at $\xi = \xi_c$, x_m goes to zero. This equation also

says that the more extended the source relative to the beam, (δ_S increasing), the smaller the fraction of the flux CLEAN can remove from a single location before moving to another. The shape of the residual and the location of the maxima as a function of ξ is given in Figure 6.2.

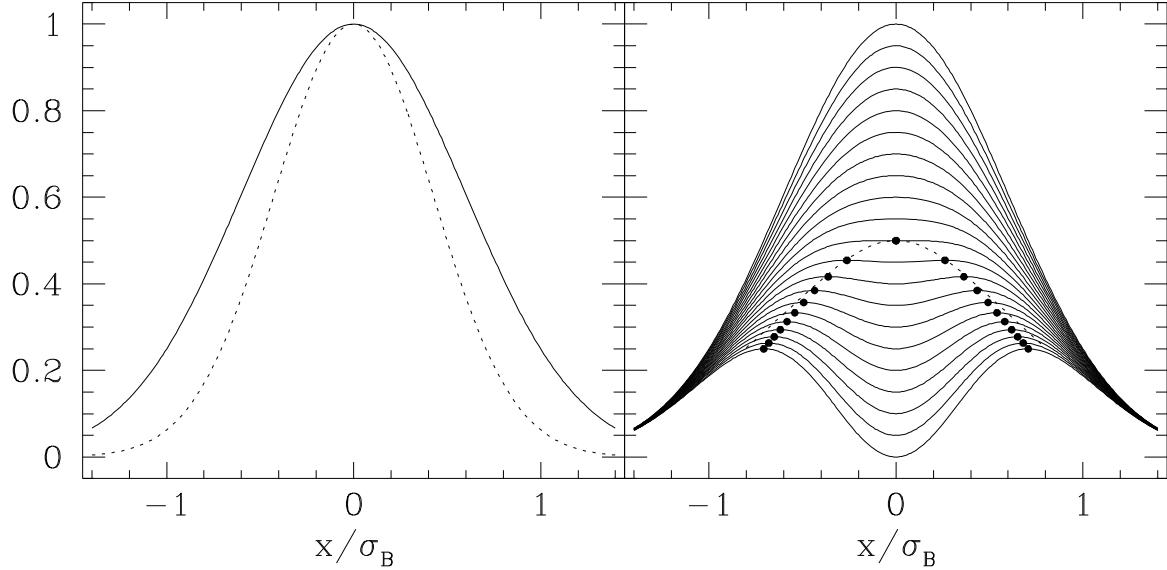


Figure 6.2: Example of initial CLEANing: The left panel shows the dirty beam and dirty image. The intrinsic size of the source is the same size as the beam. The right panel shows the residual image after various multiples of the beam have been subtracted away. The traces correspond to $\xi = 0.00, 0.05, 0.10, \dots, 1.00$, and the dot on each is the location of the maxima. The dotted line is the approximation to that position, equation 6.2. For this source size, $\delta_S = 1$ and $\xi_c = 0.5$.

A particularly simple form of the expression for the radius of maximum residual can be found by expanding ξ around ξ_c . Define $\xi \equiv \xi_c + \Delta\xi$, use $\ln(1 + x) \approx x$ and the definition of ξ_c to obtain

$$\begin{aligned} \frac{x_m}{\sigma_B} &= \left[\ln \left[\xi_c (1 + \delta_S^2) + \Delta\xi (1 + \delta_S^2) \right] \frac{(1 + \delta_S^2)}{c \delta_S^2} \right]^{1/2} \\ &\approx \left[\Delta\xi (1 + \delta_S^2) \frac{(1 + \delta_S^2)}{c \delta_S^2} \right]^{1/2} \\ &= \sqrt{\Delta\xi} \frac{(1 + \delta_S^2)}{\sqrt{c} \delta_S} \end{aligned} \quad (6.2)$$

In the limit of small overshoot then, the radius of the next component goes as the square root of the overshoot. Since $\Delta\xi \ll 1$, this has the effect of moving the radius out rapidly with overshoot, and then increasing relatively slowly. If CLEAN typically overshoots by half the loop gain times ξ_c , the appropriate ξ for use in equation 6.1 is $\xi = \xi_c + g(1 - \xi_c)/2$.

For $\delta_S = (0.1, 0.5, 1.0)$ and a loop gain $g = 0.1$, $x_m/\sigma_B = (.135, .150, .188)$. That is, the radius of the first off peak component is only a weak function of the source size. The factor of $1/\delta_S$ does not cause the expression to diverge, since $\delta_S \rightarrow 0$ also implies that $\xi \rightarrow 1$, and in that limit, the original equation 6.1 becomes

$$\frac{x_m}{\sigma_B} \approx \frac{1}{\sqrt{c}}(1 + \frac{1}{2}\delta_S^2) \approx .601(1 + \frac{1}{2}\delta_S^2)$$

Returning to the 2 dimensional case, we next present CLEAN component images where both the dirty image and beam are discretely sampled Gaussians. These images are heavily oversampled for clarity, at 14 points per beamwidth, but as verified in other tests not presented, this makes little difference in the effects shown — it just makes them easier to see. At this high oversampling, the first off peak component is predicted to be 2.1 and 2.6 pixels from the center, for $\delta_S = 0.5$ and 1.0, respectively. In Figure 6.3, the first off peak component in each panel is the nearest one to the center and is indeed two pixels out. The surprising behavior comes from the more difficult to analyze later stages of the iteration. The CLEAN components are placed in distinct rings around the peak, usually alternating in sign and spaced roughly at half widths of the dirty beam. Furthermore, while the rings get thicker when run to extreme numbers of iterations, they remain distinct and the central region about the peak never fills in. Note that this cannot be a sidelobe driven phenomena similar to that described by Schwarz, since in these simulations the dirty beam is a perfect Gaussian and *has* no sidelobes.

The radial distance of the first ring is well outside the distance predicted for the first off peak component. Figure 6.4 gives a more detailed look at the placement of the initial components. When the first component off the peak is subtracted, it still subtracts off a fraction of the central minimum as well. On the other side of the center, the approximation of two differenced Gaussians is still pretty good, and the depth of the central depression is even lower than after the initial overshoot. The second off peak component will be across the center from the first, at a radius crudely predicted by equation 6.1, but with a larger value of ξ than before implying a larger radius. The same argument applies again, (so long as the approximation of the residuals along a line connecting the center and the minimum by two differenced Gaussians holds up), and the third off peak component is out at a further distance still. Since the position directly across from the last subtraction has been flattened in the iteration before that, the third off peak residual will occur at some different position angle. The tendency for the radius to increase and to select a position angle that was not recently used will result in the components spiraling out from the center as shown in the figure.

We can go even one step further in the abuse of the two Gaussian approximation to the residuals. If one assumes that the residuals always fit this form and takes the value of the central pixel (and hence the fractional CLEAN depth ξ) as a measured value, one can predict the radius of the next CLEAN component. Figure 6.5 does exactly this, for

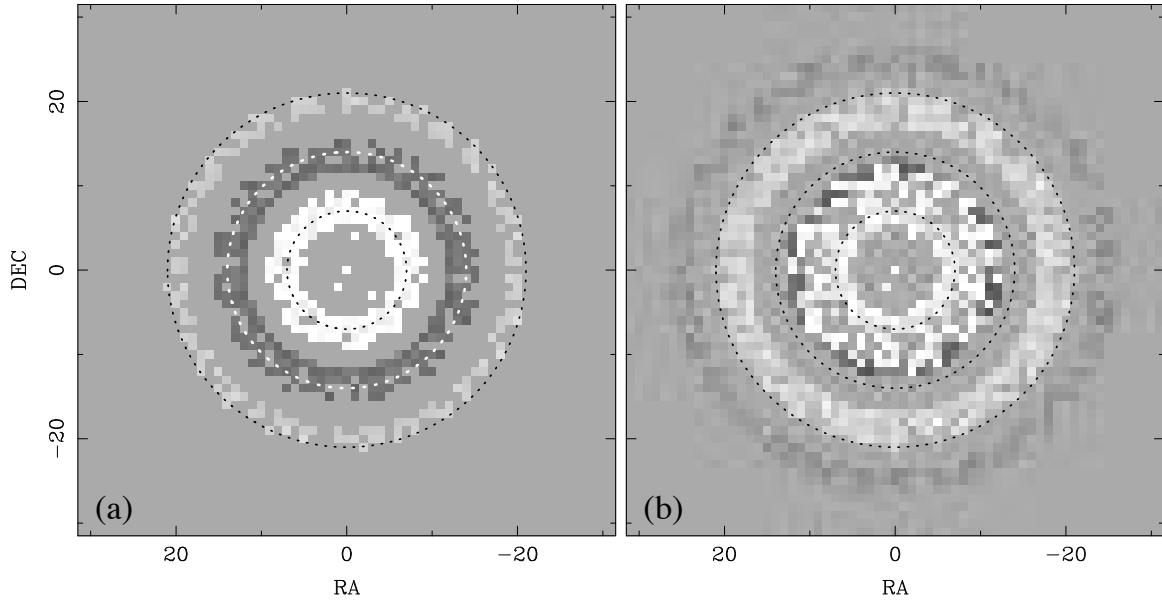


Figure 6.3: CLEAN component images. Source and beam are Gaussian with $\delta_S = 0.5$ and 1.0 for the left and right panels respectively. The left panel has been CLEANed to 2000 iterations, and the right to 5000, both with a loop gain of 0.1. The dirty beam is heavily oversampled with 14 points per beam and 1'' per pixel. The greyscale is two sided logarithmic around zero and common to both panels. The circles have radius 7, 14, and 21 pixels, and the beamsize at half maximum is equal to the smallest circle.

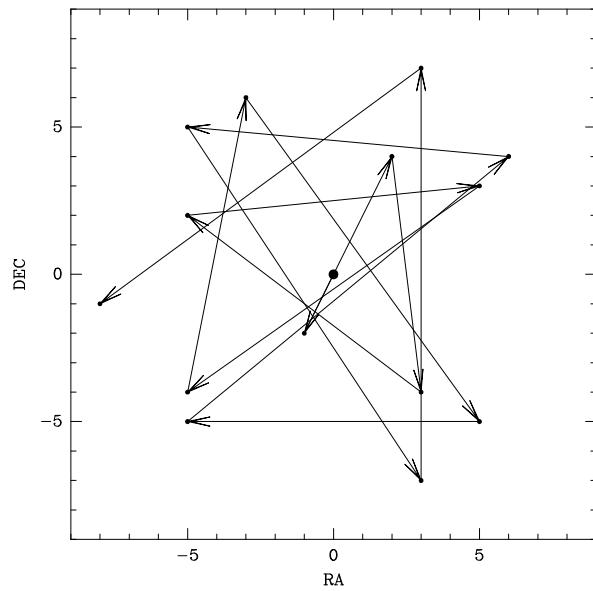


Figure 6.4: CLEAN component sequence. This plot shows the placement and sequence of the first 20 CLEAN components from Figure 6.3b, $\delta_S = 1.0$.

the same $\delta_S = 1.0$ case as before, and the agreement between the actual behavior and this semi-analytic model is surprisingly good.

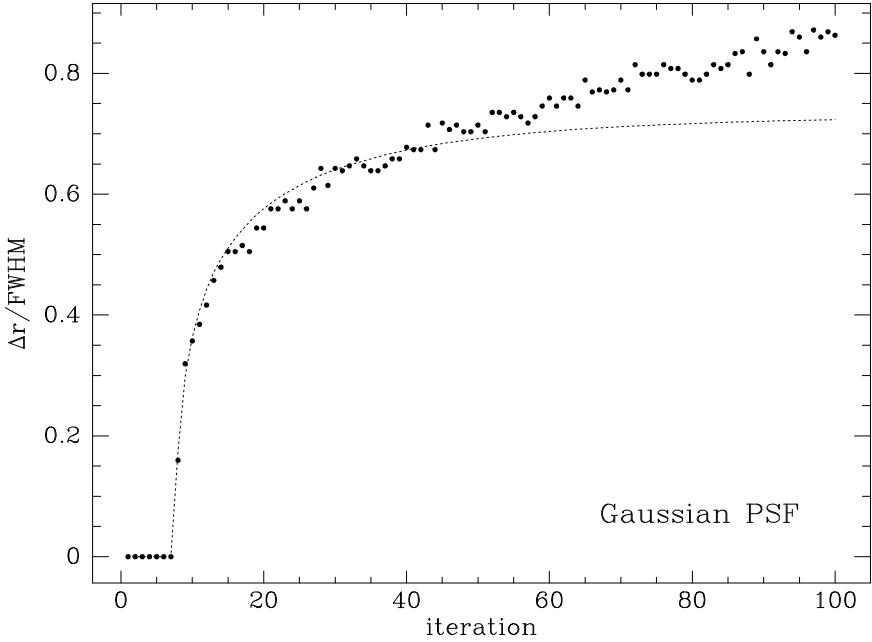


Figure 6.5: CLEAN component radial distance vs. iteration for the same case as Figure 6.4.

6.2.2 The Gaussian Approximation Applied to a VLA PSF

This is all very encouraging that the agreement between model and observed behavior is good, but the obvious point is that true point spread functions are rarely Gaussian. Figure 6.6 shows the CLEAN components for the same $\delta_S = 1.0$ case as before, but this time with the same uniformly weighted VLA Full Track (3C48) data used so frequently before. (Observation and imaging parameters are given in Table 3.7 and the PSF is shown in Figure 3.14.) The source is an elliptical Gaussian the size of the fitted beam. The cell size is $0.03''$ in the first panel and $0.015''$ in the second, giving 6.5 and 13 points per beam. The weighting field of view is 1.0 in the first image and 2.0 in the second, resulting in identical gridded weights after uniform weighting. Again we see the CLEAN components clustering in rings of positives and negatives, even though the rings are broken due to the azimuthal dependence of the real PSF. The central region about the peak again does not fill in, even in the limit of extreme CLEANing. Finally, note that there is essentially a one to one correspondence between clusters of components. Going from a high to an extreme oversampling has made essentially no difference in where the components are placed.

The semi-analytical model predicting CLEAN component radius from the central pixel value of the previous iteration is given in Figure 6.7. The fit is still remarkably

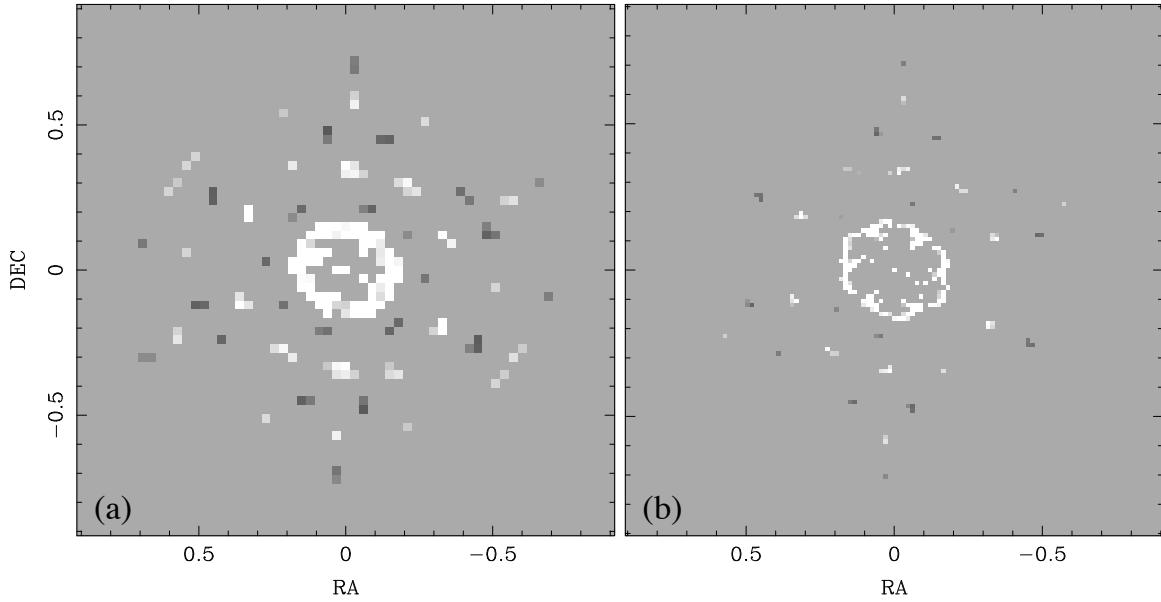


Figure 6.6: CLEAN component images for a real VLA beam and Gaussian model source. The only difference between the two panels is that the pixel size has been halved from panel a to b. The cell size makes little difference in the placement of the clusters. The greyscale is two sided logarithmic around zero and common to both panels.

good given the crudity of the residual model, but now only qualitatively correct rather than quantitatively so. There are two traces due to the fact that it's no longer clear how δ_S should be calculated. Recall that $\delta_S = \sigma_S/\sigma_B$. The problem is that the fitted beam analytically convolved with the model no longer equals the fitted dirty map. The magnitude of the effect is shown in Figure 6.8. If we compare the fitted beam with the model, we obtain $\delta_S = 1.0$ as before. But it's arguably the dirty image which is more important for the residuals. Fitting a Gaussian to the dirty image and working backwards, one obtains $\delta_S = 0.799$. In Figure 6.7 we simply take the pragmatic approach and note that both values give a qualitatively reasonable fit to the data.

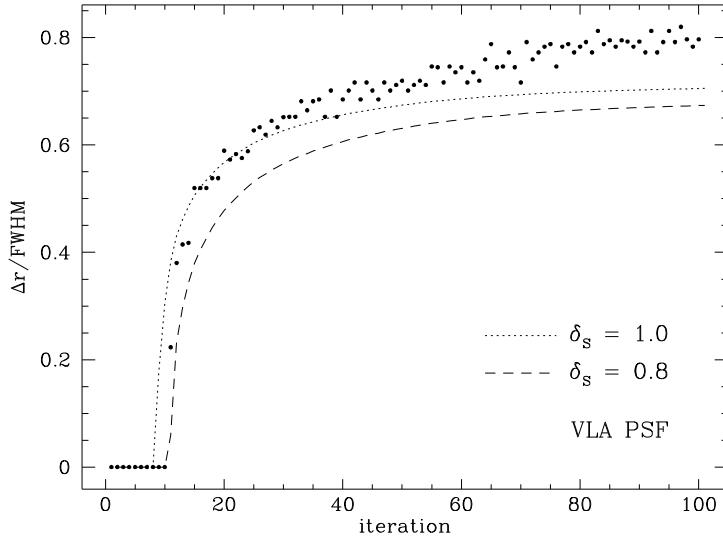


Figure 6.7: CLEAN component radial distance vs. iteration for VLA PSF case of Figure 6.6. The two dotted traces are for the actual $\delta_S = 1.0$ used in the modelling and the fitted $\delta_S = 0.8$, which fits the main dirty lobe better.

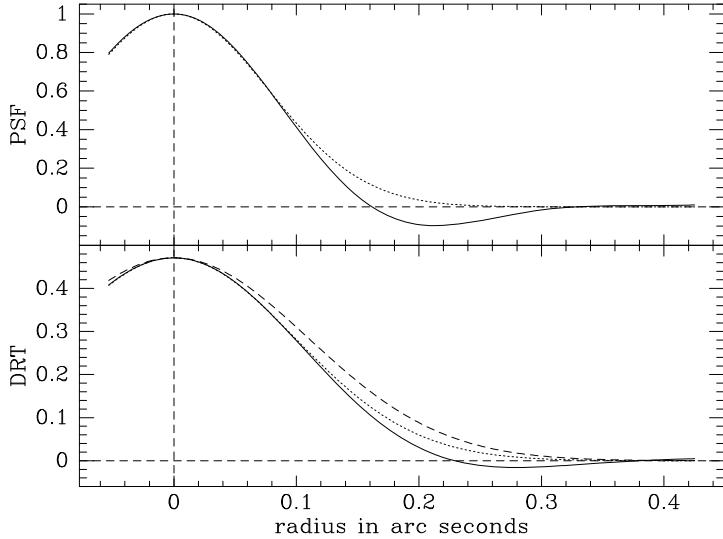


Figure 6.8: Convolution behavior of the VLA PSF. Panels are the minor axis slice along the PSF and dirty image. The light dotted line is the fitted Gaussian to each, and the heavy dashed line is the convolution of the fitted Gaussian to the PSF with the Gaussian model. Notice how the negatives in the PSF have created a significantly narrower dirty image than predicted analytically.

6.2.3 Multiple Orders of Clean Components

The crude analytic approximation makes no mention of multiple orders in the clean component spacing at all, much less the spacing between orders. Here we will do little better, and merely note some suggestive observations; more work could certainly be done on this topic. First of all, we note that while the bulk of the flux is in the central spike and first ring, the total CLEANed flux in the outer rings is significant. For instance, returning to Figure 6.3, the flux contained in the inner 4 components, first, second and third rings of the $\delta_S = 0.5$ case is $(.5847, .4796, -.0821, .0179)$ Jy, respectively. The summed flux from these orders are $(.5847, 1.0643, .9822, 1.0001)$ Jy, so the total flux oscillates around the correct as more orders are included. In the wider model $\sigma_S = 1.0$ case, the flux in the orders is $(.2067, .7379, .0647, -.0099)$ Jy and the summed flux is $(.2067, .9946, 1.0093, .9994)$. In the later case, only the first two components have been counted as part of the central spike. This order behavior is probably responsible for the tendency of CLEAN to systematically over-estimate, then under-estimate, the source flux as the CLEAN progresses. New orders are added only when the orders interior to it are substantially completed in terms of the flux they contain. If one stops the iteration when the final significant order is a positive one, the estimated total flux will likely be slightly high. If the last ring is negative, the estimated flux will be low. A similar behavior will be seen if one CLEANs very deeply and then estimates the flux by summing the components within some arbitrary radius of the peak — the estimated flux will oscillate around the correct value as a function of integration radius.

It would be nice to understand the spacing of these orders in detail, but as yet this is an open question. The order spacing for the $\delta_S = 0.25$ case is nearly identical to that of the $\delta_S = 0.5$ case, yet both are very different from that of the $\delta_S = 1.0$ case. In the small source limit, the spacing seems not terribly different from one half of the beam width or a bit less, and it seems to become slightly smaller than that as the source size increases. Estimating the ring spacing is made more complicated by the fact that the width of the orders grows with the source size. For a fixed source, the width of the ring decreases with radius, and one might clean extremely deeply and estimate the spacing from the outermost rings.

One might expect that the distance between maxima in the residual model with central depression zero would be a characteristic spacing of this problem. This corresponds to the $\xi = 1.0$ trace of Figure 6.2, and the width is given by equation 6.2.1. Unfortunately the limit of .601 times the beamwidth does not match any measured scale. Another obvious possibility is that the loop gain might influence the spacing of the rings. But cutting the loop gain by a factor of two or four and increasing the number of components by the same factor yields an almost unchanged component image. The only difference is in how quickly the components spiral out to the first ring. There is no dependence of either the spacing or width of the rings on loop gain. The central region never fills in, even with exceedingly small loop gain.

In Figure 6.9, we show an expanded version of the radius against iteration plots. Even more clearly than the component images, this shows the presence of the multiple orders. Note that in Figure 6.3, the inner most ring is quite wide and predominantly positive, with a sharply defined negative outer edge. The upper panel of Figure 6.9 shows this to be the merger of two otherwise distinct orders. The innermost positive ring has merely grown wide enough to partially obscure its negative neighbor. Additionally, we can see that at any given point in the iteration there is a clearly defined radius for each order. Typically the ring starts or moves outwards quite abruptly, and then gradually spirals back inwards resulting in the finite width. Tabulating the limiting order spacing on plots like these against δ_S might provide sufficient leverage to unravel the underlying mechanism in future work.

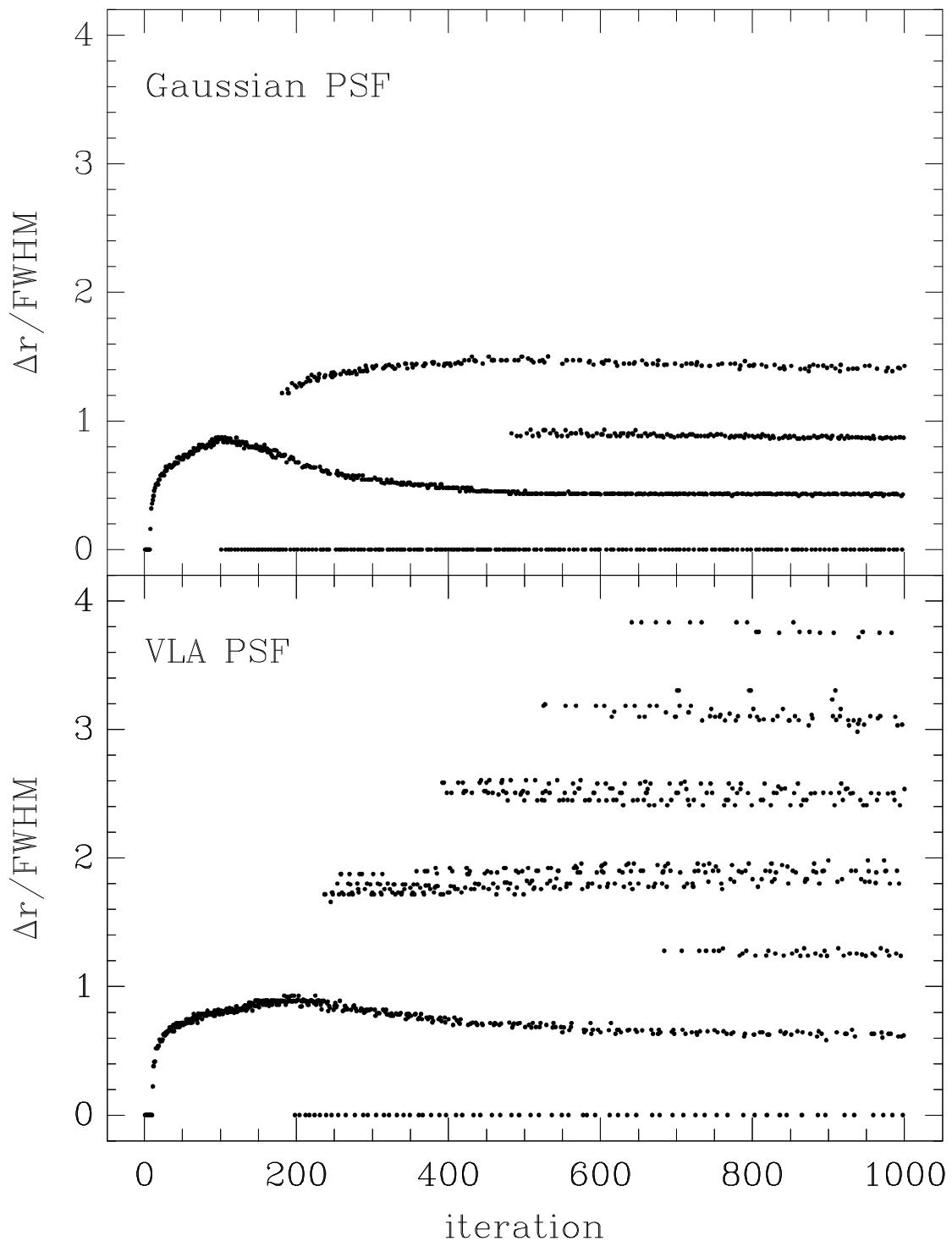


Figure 6.9: CLEAN component radial distance vs. iteration. These are the same quantities as Figures 6.5 and 6.7, but with a much extended iteration axis. This clearly shows the breakup of the traces into multiple orders and the morphological similarity of the plots for a Gaussian PSF and VLA PSF.

6.2.4 Fitting to CLEANed Gaussians

If working directly with the CLEAN components is a bad idea for estimating flux, and if one cannot fit directly to the visibility data due to confusion, one must work with the restored image. This of course introduces the complication of what restoring beam to use. In the simple simulation set presented here, an elliptical Gaussian model was fit to the CLEANed image of an isolated noise-corrupted Gaussian, using all points greater than .35 of the image peak as input to the fit. Using a larger restoring beam will include more points in the fit, reducing the thermal noise but will make the deconvolution more ill conditioned. The same VLA PSF was used as before, in the extreme oversampled limit of 14 points per beam. For each of the parameters $\delta_S = 1.0, 0.5, 0.0$, a database was created with SNR of 50 and 10, along with a noiseless database. Here SNR is the peak response in the dirty map over the RMS thermal noise. No calibration errors were introduced. Each combination of parameters was CLEANed to 1000 components, which most scientists would consider too many, and also to a “reasonable” stopping point which was determined by eye as the point when the flux stopped changing significantly. For all of these, the fitted flux of the Gaussian and the mean beamwidth was tabulated as a function of the size of the restoring beam. Only one realization of the thermal noise was used for each case — there was no attempt at a Monte Carlo style simulation. A Gaussian PSF case was substituted for the infinite SNR case with “eyeballed” stopping criterion, since it is difficult to over CLEAN noiseless data. This provided a best case lower limit on the error for each major category. And finally, a single $\delta_S = 0.25$ case was substituted in the point source case where the VLA PSF was already producing a perfect answer and there was no need for the Gaussian lower limit. The results of these simulations are given in graphical form in Figure 6.10. Sizes are read from the left vertical axis of each panel, and fluxes from the right axis. The results are normalized such that a width of δ_S and a flux of unity is the correct answer, and this is marked by a solid diamond on the appropriate axis.

The basic results are that a restoring beam of the same size as the fitted beam is a reasonable choice. It is not necessarily the best choice for any given case, but it is not too far off and there is no other single better choice. In the noisy cases, both the fitted flux and the fitted size were systematically underestimated. Needless to say, anyone who wishes to do photometric statistics on CLEANed objects would be well advised to do much more extensive modelling than this, but I hope this series gives an idea of the magnitude of the errors which might be encountered. Also, see (Cohen, 1991) for a simulation series of MEM photometry on point sources in a crowded field. While addressing a somewhat different problem than this one, her general result of systematic flux underestimation is similar to the results here.

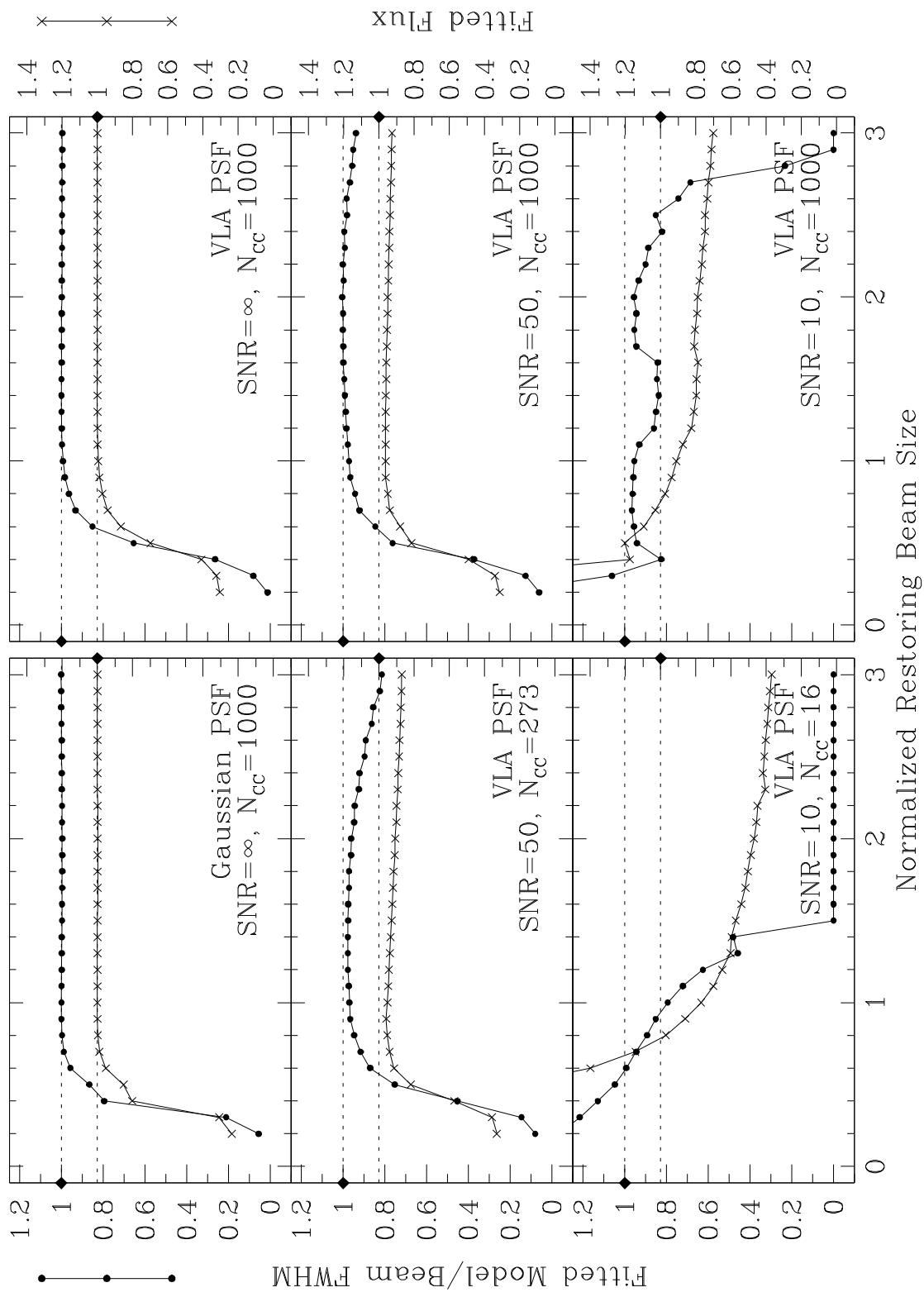


Figure 6.10a: CLEANed Gaussian Model Fits, $\delta_S = 1.0$. See text for details.

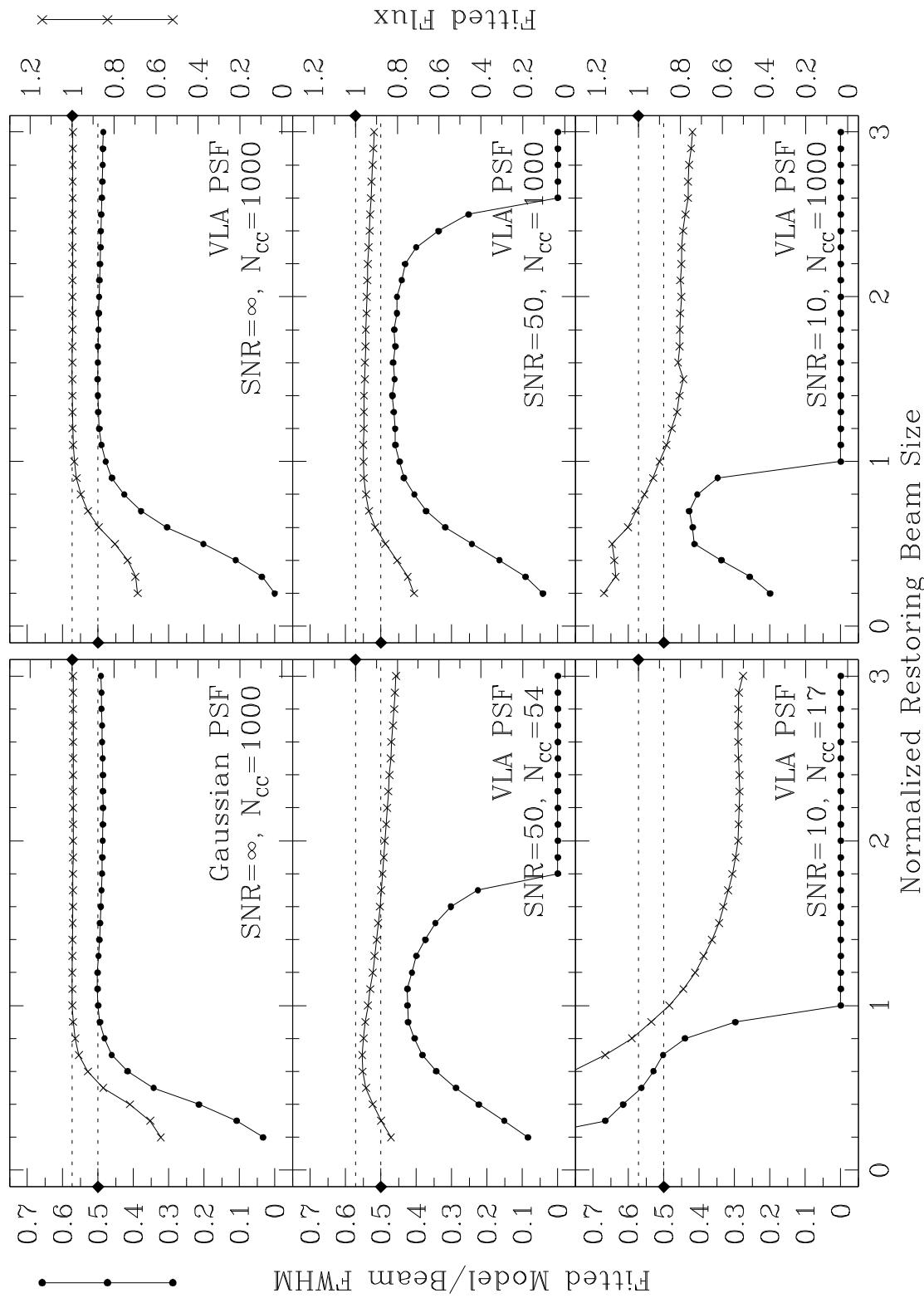


Figure 6.10b: CLEANed Gaussian Model Fits, $\delta_S = 0.5$

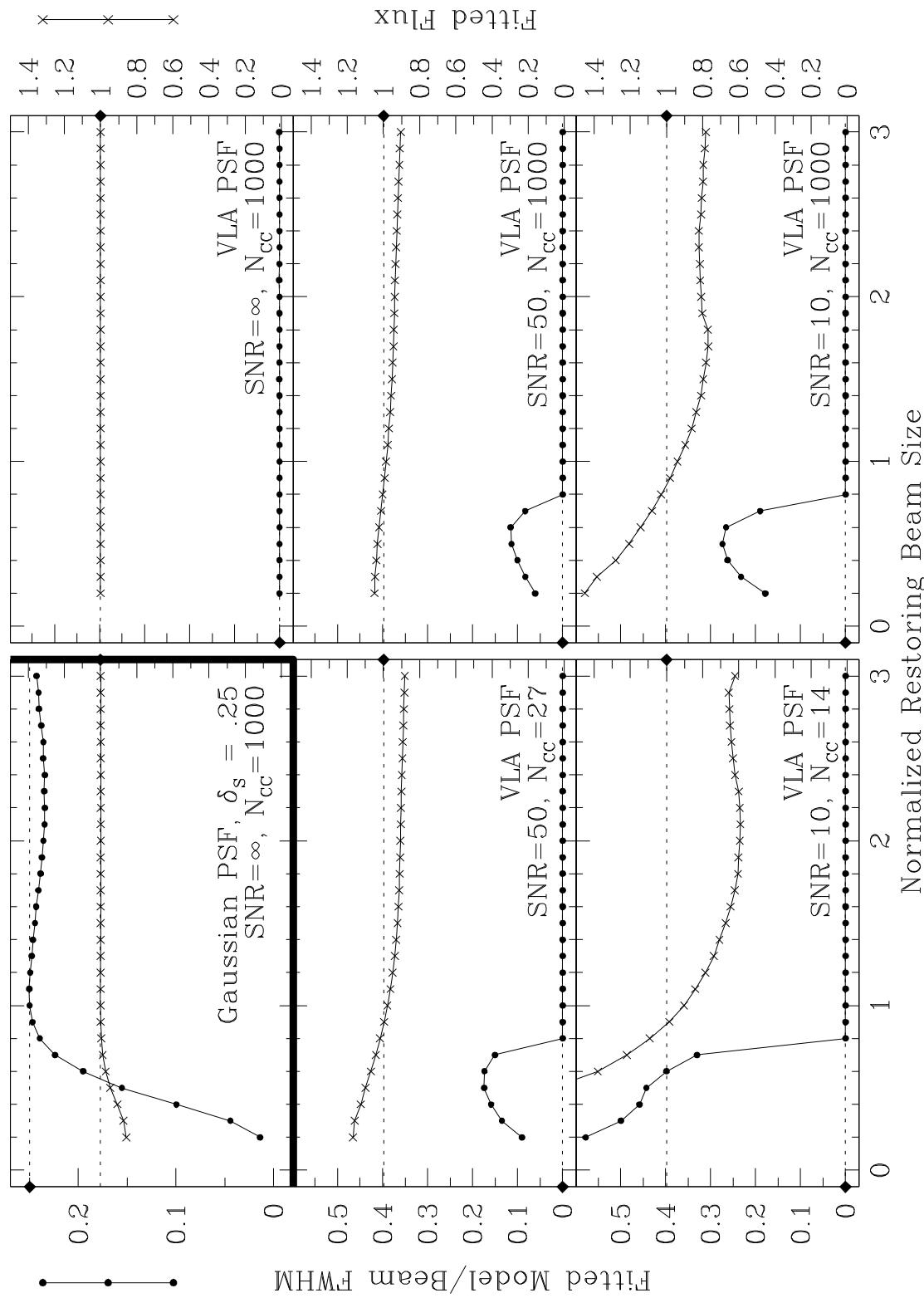


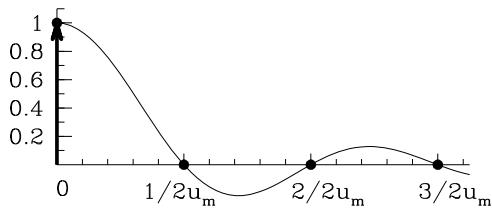
Figure 6.10c: CLEANed Gaussian Model Fits, $\delta_S = 0.0$ except where noted.

6.3 CLEAN components are Fourier Components in the Limit

Since the work of Schwarz, (1978), it has been known that CLEAN components are a fit to the data in the Fourier domain, in a least squares sense. It is also fairly easy to show that to minimize the L_2 norm between a real function and a *finite* Fourier series of given length N , the coefficients must be the first N coefficients of the standard *infinite* Fourier series. See for example, Arfken (p. 646, 1970). In this light, it is not terribly surprising that CLEAN components become Fourier components in some limit. In a sense the manipulations in this section are really quite trivial, but this is an important enough point that I provide an explicit alternative derivation working primarily in the image plane.

As before, S is the sampling function. We specialize to the one-dimensional perfect sampling case, where

$$S(u) \equiv 1/2u_m \Pi(u/2u_m) = \begin{cases} 1/2u_m & |u| < u_m \\ 0 & \text{otherwise} \end{cases}$$



The continuous dirty beam is

$$B = \mathfrak{F}[S] = \text{sinc}(2u_m\ell)$$

In discretizing the problem, we specialize again to a critically sampled image. This is an image with cell size $\Delta\ell = 1/2\hat{u}_m = 1/2u_m$. The gridded u - v

plane exactly contains the measured data and exactly satisfies the sampling requirements of the dirty beam. Other than the central peak, the samples of the beam fall exactly at the zeros of the sinc function. We can easily visualize the CLEANing process with such a beam and see that each pixel in the residuals can be reduced to zero independently of its neighbors. Alternatively in matrix notation, the beam is simply the identity matrix. The CLEAN process is simply the solution of

$$\mathbf{r} = \mathbf{d} - \mathbf{B}\mathbf{c}, \quad \text{where } \|\mathbf{r}\|_2 \text{ is to be minimized.}$$

Given the trivial nature of the beam, we can immediately write down an exact solution with $\mathbf{r} = \mathbf{0}$, namely $\mathbf{c} = \mathbf{d}$. In this case, the value of the component representation is merely the value of the sampled dirty image at that location.² Notice that algorithms such as MEM which only minimize the residuals to within the noise are not guaranteed to return the solution $\mathbf{c} = \mathbf{d}$, even with such a beam as this. Writing down the continuous dirty map, the explicit component representation becomes

$$c_n = \mathbf{c}(n\Delta\ell) = \int_{-\infty}^{\infty} I(\ell) \text{sinc}(2u_m(n\Delta\ell - \ell)) d\ell,$$

²This ignores aliasing from the finite size of the image. For the purposes of the demonstration, the image may simply be made “large enough” that the effects are negligible.

where $\mathbf{c}(n\Delta\ell)$ means the component of \mathbf{c} corresponding to the spatial position $n\Delta\ell$. In Appendix F we show that an arbitrary finite portion of the visibility function can be expanded as

$$V(u) = \sum_{n=-\infty}^{\infty} a_n e^{in\theta}$$

where $a_n = \frac{1}{2u_m} \int_{-u_m}^{u_m} e^{-in\pi u/u_m} V(u) du$

When we substitute the fundamental Fourier transform relationship equation 2.7 into this expansion, this becomes

$$a_n = \frac{1}{2u_m} \int_{-u_m}^{u_m} e^{-in\pi u/u_m} \int_{-\infty}^{\infty} I(\ell) e^{-2\pi i u \ell} du$$

The infinite integral will converge by virtue of the implicit antenna directivity, so we may rearrange to

$$= \frac{1}{2u_m} \int_{-\infty}^{\infty} I(\ell) \left[\int_{-u_m}^{u_m} e^{2\pi i u (-n\Delta\ell - \ell)} du \right] d\ell$$

Since

$$\int_{-u_m}^{u_m} e^{iau} du = \frac{1}{ia} [e^{iau_m} - e^{-iau_m}] = \frac{2}{a} \sin(au_m)$$

$$\begin{aligned} a_n &= \frac{1}{2u_m} \int_{-\infty}^{\infty} I(\ell) \frac{2}{2\pi(-n\Delta\ell - \ell)} \sin(2\pi(-n\Delta\ell - \ell)u_m) d\ell \\ &= \int_{-\infty}^{\infty} I(\ell) \operatorname{sinc}(2u_m(-n\Delta\ell - \ell)) d\ell \\ &= \mathbf{c}(-n\Delta\ell) = c_{-n} \end{aligned}$$

Thus the CLEAN component representation on the sky has a one-to-one correspondence with the coefficients in the complex Fourier decomposition of $V(u)$. This immediately has several consequences, as we can now utilize the extensive body of knowledge about Fourier series. The most important consequence is that the CLEAN component representation is complete in this limit — there is no possible source that cannot be represented by CLEAN components. But being coefficients of a Fourier series, there is no particular reason that the normal constraints of bounded support and positivity should be satisfied. In fact, while applying such constraints generally improves the solution in practice, it is also clearly the *wrong* thing to do from the point of completeness. We sacrifice mathematical completeness in the name of pragmatically introducing approximate a priori information. There are theorems about the magnitude of Fourier components decreasing like $1/n$ when applied to regions containing step function discontinuities, (Raisbeck, 1955), but attempts to apply this prediction to real problems containing shifted point sources have not been successful. The differences between real beams and the idealized one above are just too great. The importance of this result is mostly philosophical in nature — CLEAN components represent the visibility data, not the sampled sky.

6.4 Model Combination in the Fourier and Image Planes

Finally, we conclude the chapter with an example of the importance of high spatial frequency extrapolation in the u - v plane and a few general comments on model combination. Recall from Section 5.2 that MEM generally does a better job at high spatial frequency recovery than CLEAN, though it does have problems with extremely compact emission. By contrast, CLEAN generally does a slightly better job at total flux recovery and the very lowest spatial frequencies. It is clear that the solutions in the different regimes of the u - v plane may have quite different characters. One is tempted to try and combine the models produced by different deconvolution algorithms, using the best region of reconstruction from both. We do this routinely when subtracting off CLEAN components for compact emission and deconvolving the residuals with MEM. This is a generally done in the image plane and in a sequential manner, allowing the second algorithm working on the residuals to correct whatever it considers a mistake made by the first algorithm. One would not generally perform two independent deconvolutions of the entire data set and then combine the components piecewise in the image plane. An obvious problem is that there would be no guarantee that such a patchwork quilt of model components would even satisfy the original convolution equation. However, by combining the models in the u - v plane instead of the image plane, nearly this exact prescription works quite nicely. It may or may not produce a better image, but at the least it will do essentially as well as the individual algorithms.

Figure 6.11 is yet another deconvolution series of the model 3C48 source given in Table 3.8. In this case there are only the 4 main Gaussian components and no outlying point sources. The model was discretized in the image plane and convolved with the discrete uniformly weighted beam from Figure 3.14, so the only source of error in these plots is coming from failures in the deconvolution. The top two panels are standard CLEAN and MEM reconstructions of the source, both tightly boxed to the support of the model. The CLEAN used a loop gain of 0.1 and reduced the residuals to 5 μ Jy. The MEM used a `sigma` of 10^{-5} , a small initial flux estimate and a flat default. It was somewhat arbitrarily run for 1000 iterations, the large number being needed due to the compact structure of the nearly unresolved Gaussian. (The MEM reconstruction is still improving at iteration 1000, but *very* slowly.)

To investigate u - v plane model combination, the SDE task `xfrstat` was used to automatically generate the envelope of the sampling pattern in the u - v plane, and this envelope was scaled by 0.9. All points interior to the rescaled envelope were set to unity in a weighting mask, and the sharp edges in the mask softened by smoothing in the u - v plane with a circular Gaussian of half width 10% of the maximum u - v spacing and unit integral. Composite models were generated by forming

$$MOD_{comp} = \mathfrak{F}^{-1} \left[W \mathfrak{F}[MOD_{inner}] + (1 - W) \mathfrak{F}[MOD_{outer}] \right],$$

where W is the weighting mask just described. Combining the CLEAN and MEM models in this way resulted in the lower two panels of Figure 6.11. The most striking feature of these

two panels is how much the composite images resemble that of the algorithm contributing the highest spatial frequencies. Using CLEAN for the inner portion of the $u-v$ plane has improved the pure MEM image slightly, at cost in worse negatives, but in most respects the composite image is still inferior to the pure CLEAN image. Using MEM to generate low spatial frequencies for CLEAN, has indeed smoothed the composite image somewhat, and improved the worst negative a bit, but at cost in image artifacts and a slight increase in off source RMS. For this particular case there is no improvement from $u-v$ plane model combination, but it is an interesting technique and might someday prove useful.

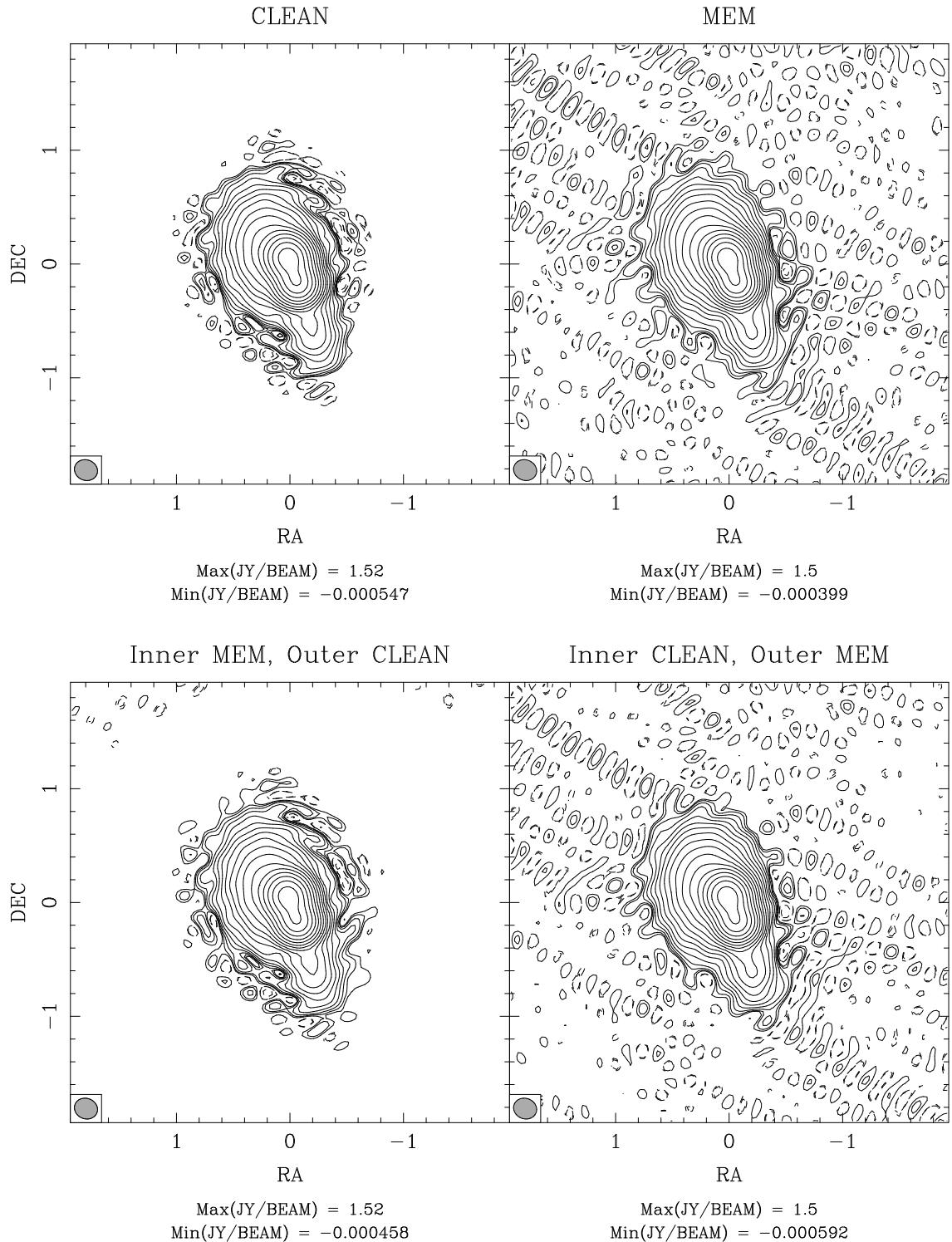


Figure 6.11: Combination of CLEAN & MEM models in the u - v plane. 4 contours per decade, lowest contour is $100 \mu\text{Jy}$.

In Figure 6.12 the problem has been tackled with the more traditional approach of CLEANing the compact structure lightly and then using MEM on the residuals. It hasn't helped, and in most respects has simply made the reconstruction worse. The problem is coming from the especially challenging nature of the source. As shown in Chapter 5, while deconvolution errors at this level are to be expected even for the simplest of sources, proper choice of algorithm should be able to reduce the error on any given size scale to better than we are doing here. So what is wrong?

The problem here is that the two most compact components appear similar, but are in fact quite different from the point of view of the deconvolution algorithms. The northern compact component with the larger peak is extended, with a minor axis 44% of the mean beamwidth. By contrast, the sharper southern compact component has a minor axis only 13% of the mean beamwidth. This makes all the difference in the ability of MEM to deconvolve it well. When MEM alone is run on the model, the center of the error pattern is the sharp feature. When CLEAN alone is run on the model, the center of the error pattern is the extended peak. Unfortunately, there is no single threshold where CLEAN will remove just the flux from the sharp peak. In fact, it will start removing flux from the extended but stronger peak, and by the time it has removed enough flux to significantly help the MEM algorithm, it will already have embedded the characteristic CLEAN errors on diffuse peaks into the solution. For such delicate problems as these, one would want a CLEAN that selects components based on the difference of the residuals and a smooth model of the extended emission — not unlike Multi-Resolution Clean (Wakker and Schwarz, 1988) or the algorithm proposed by Dwarakanath et al. (1990). Tests trying to crudely model such a procedure with extremely tight CLEAN boxes were not successful, so it is not known how well such a procedure would work in this demanding case. In any event, the combination of two algorithms roughly doubles the number of free parameters that must be set in the reconstruction process, to say nothing of the modelling required to separate the components into completely unresolved and marginally resolved categories. Even if particularly careful and sophisticated use of existing tools could be made to significantly improve the deconvolution of this source, there is no question that the NNLS algorithm of Chapter 4 is providing a much better reconstruction with essentially no arbitrary parameters beyond the rough source support. Methods such as model combination need only be used when the source size is too large for NNLS based methods, and in such cases, an adequate model of the source to discriminate between resolutions might be prohibitively difficult.

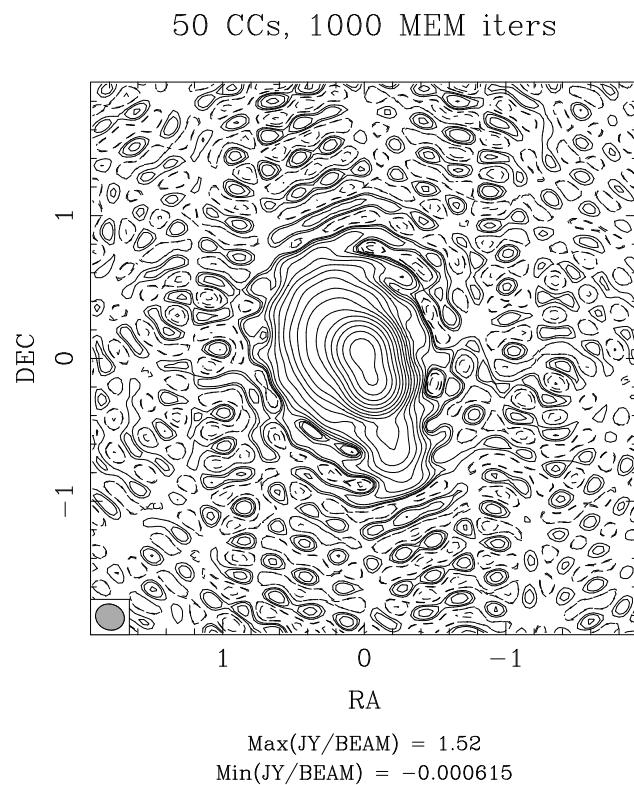


Figure 6.12: Combination of CLEAN & MEM, run sequentially in the image plane. The dirty image was lightly CLEANed with 50 components, followed by 1000 MEM iterations on the CLEAN residuals. 4 contours per decade, lowest contour is $100 \mu\text{Jy}$.

Chapter 7

Case Study: 3C48

7.1 Introduction

The initial impetus for much of this dissertation came from an observation of 3C48 by Craig Walker that was unaccountably limited to a dynamic range of some 16000:1, significantly above the thermal noise.¹ A simple image plane model of the source was constructed and the deconvolution process simulated. The venerable Högbom CLEAN produced a peak-to-RMS dynamic range of over 100,000:1 on a noiseless, discretized, perfectly-calibrated realization of this model, but also produced highly visible artifacts and negatives at the 3000:1 level. This provided the first evidence that the deconvolution process was a limiting factor comparable to residual calibration errors for high-precision VLA observations. Moreover, the morphology of the error pattern closely matched that seen in the observational image, although reduced in amplitude. In subsequent work it was realized that part of the problem with the real data *was* in fact residual calibration error and the neglect of an offset confusing source. Once the data were well calibrated, the correspondence between the model and observational error pattern became striking, both qualitatively and quantitatively.

At the observation frequency of 8.4 GHz and using the high resolution VLA A configuration, 3C48 is a fairly compact source, with two nearly unresolved components and a modest extended halo. Prior to this dissertation, such a source was considered close to ideal for CLEAN, so the deconvolution failure was surprising. Other deconvolution algorithms, including MEM, SDI CLEAN, and Multi-Resolution CLEAN, fared little better. (The particular implementations used are described in Section 2.7.) Eventually it was discovered that a similar error morphology was produced with a single isolated Gaussian model, and this led to the resolution tests described in Chapter 5. Still later, it was found that the particular combination of size scales present in the two compact components did present a challenge more difficult than any single isolated component. This was described briefly in the context of model combination in Section 6.4. No significant progress was made in the deconvolution of the 3C48 model source until the advent of the NNLS algorithm described in Chapter 4. NNLS is the only algorithm studied which was capable of reducing the deconvolution artifacts on the source below the level of the thermal noise. This chapter

¹The factor of “10 above the thermal noise” given in Briggs and Cornwell (1994) is quite erroneous.

is somewhat anti-climactic, since we have already used models of this source and observation as test cases in many other contexts. Here we simply perform a reasonably careful noise analysis of the observation, present the final observational images at X band and L band, and briefly describe their astrophysical significance.

7.2 The Data and the Noise

Interleaved X and L band observations of 3C48 and calibrators were made on August 21, 1991, spanning a period of approximately 13 hours. Of that, 5.5 hours of good on-source data at X band were obtained for an after-calibration total of 110338 visibilities at an integration time of 60 seconds, averaged down from the original 10 second data. The L band data was left at the original time resolution, and 619187 visibility records were retained after calibration and flagging, for an equivalent on-source time of 4.9 hours. 3C48 has a declination of 33.2° so the $u-v$ coverage was excellent. The X band coverage is given back in Figure 3.12, and the associated point spread functions in Figure 3.14. The L band data has coverage and PSFs that are visually nearly identical to the X band data, and they are not shown. The X band visibility amplitude plot is given in Figure 7.1. The point to note is that there is quite significant structure at all size scales, and that the signal-to-noise ratio is quite high.

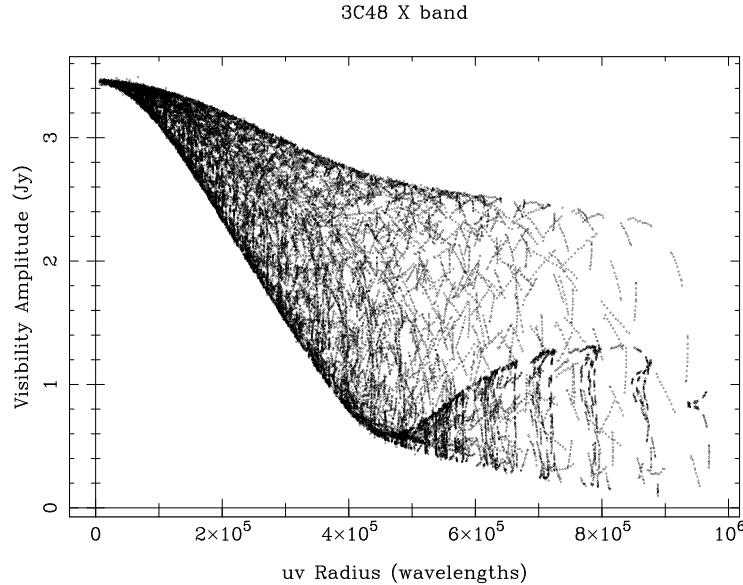


Figure 7.1: Visibility amplitude plot. The visibilities have been time averaged to 2 minutes in this plot, and all points are shown.

It must be admitted that the a priori predicted thermal noise for this observation is not a particularly good match to the observed number, but it is worth while going through the calculations in some detail to show how one might attempt as good an a priori estimate

as possible. The VLA Observational Status Summary gives the following expression for the expected map plane naturally weighted thermal noise:

$$S_{RMS}(\text{mJy}) = K \sqrt{N(N - 1)(n\Delta t_{hrs}\Delta\nu_{MHz})} \quad (7.1)$$

where K is a measured system constant and is 5.6 mJy both for the current VLA and at the time of observation. N is the number of antennas, n is the number of intermediate frequencies recorded,² Δt_{hrs} is the observation time in hours, and $\Delta\nu_{MHz}$ is the observation bandwidth in MHz.

For the purposes of careful a priori noise estimation, it is more useful to have this equation in a different form. First, note that $N(N - 1)$ is twice the number of baselines, and that $N_{base}\Delta t$ is the equivalent total time on source for a single baseline. Since data can be flagged for any number of reasons, and we generally do not conveniently know exactly how long the array was taking good data on the source, this product is better written as $N_{int}\Delta t_{int}/3600$. N_{int} is the total number of integrations, and is usually reported by mapping programs. Δt_{int} is the single integration time, now specified in seconds. Equation 7.1 becomes

$$S_{RMS}(\text{mJy}) = K \sqrt{\frac{2n}{3600} N_{int}\Delta t_{int}\Delta\nu_{MHz}} \quad (7.2)$$

Now, when a subset of the array happens to drop out for a period, the data it generates will be flagged bad somewhere in the reduction. By the time the data reaches the mapping program, the total number of good integrations will reflect that the flagged data is not present, as will the estimated thermal noise. If one wishes to take the trouble, one can even take into account that not all visibilities will have the same weights after time averaging. The weight and actual time spacing of the averaged visibility data can usually be determined from a time-baseline listing of the visibilities. A histogram of the visibility weights can often help sort out confusing situations. `uvmap` within SDE has this capability, albeit not conveniently. Once the weight associated with the nominal averaged integration time is determined, just sum the visibility weights and divide by the nominal single integration weight. The result is the effective number of full weight integration times to be used in a sensitivity calculation.

In this particular case, the data have been averaged down to 60 seconds from the original 10 seconds. Weights are present in the data file ranging from 22.96 to 160.72. The histogram shows the vast bulk of them to be 137.78, and the histogram is regularly spaced at intervals of 22.96. The highest weight in the file appears to be the visibility averaging program occasionally averaging 7 points together instead of 6. This behavior can happen when the specified averaging time is an exact multiple of the original integration time. The

²For most instruments, the number of intermediate frequencies is just the number of polarizations. The VLA has the ability to observe at two different frequencies within the same band, for a total of four intermediate frequencies.

total weight sum is $1.462(10)^7$, yielding an effective 106200 integrations of full 60s weight, instead of the reported 110400. Since the inverse RMS thermal noise scales with the square root of this quantity, 106 vs. 110 is hardly a significant effect — but it might be worth doing rather than wondering how significant it is.

The observation was done in spectral line mode, as are nearly all high dynamic range observations. See Perley (1989) for details on why this is so important from a hardware perspective; the most important reason is that it avoids nonclosing phase errors of a few degrees introduced by the analog phase-shifter in the correlator. In spectral line mode, the equivalent phase shift is done digitally in the transform domain. Several other effects become important as the bandwidth of the observation grows, most notably errors due incorrect antenna delay settings. In spectral line mode, the effective bandwidth for this error is the width of each individual channel rather than the total observing bandwidth. Finally, spectral line mode allows the bandpass to be calibrated against frequency, after which the filter-corrupted channels on the edges of the bandpass can be discarded. Spectral line mode improves the fidelity of the VLA tremendously, but at significant cost in sensitivity. In order to obtain sufficient flexibility for a high quality bandpass calibration, 32 channels were used in this experiment and the individual channel width was forced by hardware limitations to only 391 kHz. After dropping two channels from each end of the bandpass and ignoring the channel sum, 27 channels were left, for a total effective bandwidth of 10.55 MHz with both RR and LL observed polarizations. Compared to the 100 MHz obtainable in continuum mode, we have sacrificed nearly a factor of ten in usable bandwidth to get the benefits of the spectral line system. This is more than a factor of three in thermal sensitivity!

We now use equation 7.2 to estimate the expected naturally weighted thermal noise, remembering that $n = 2$ for the two polarizations. The result is that the expected map plane naturally weighted thermal RMS is $20.3 \mu\text{Jy}/\text{beam}$. Unfortunately, this does not agree particularly well with the noise estimated from the scatter in the visibility data, via the techniques of Section 6.1. The task `visavg` using successive differences mode and a threshold of 150 seconds predicts a ΔS_{uw} of .1227 Jy for a visibility with unit weight. `uvmap` predicts that the naturally weighted thermal RMS will be $2.614(10)^{-4}\Delta S_{uw}$, which produces a combined prediction of $32.1 \mu\text{Jy}/\text{beam}$ — more than 50% higher than the a priori estimate. The estimate of ΔS_{uw} was checked in a number of ways. Independent quartiles of the data, binned based on $u-v$ radius, yielded .1185, .1206, .1255 and .1258 Jy. The best model found by NNLS deconvolution was subtracted off the data to prevent bias from influencing the estimate, and the same quartiles yielded .1183, .1195, .1233, and .1216 Jy. Time bin averaging over 300 seconds yielded .1332 Jy, but decreased to .1216 Jy when the model was subtracted off first. In short, the internal estimates of the noise were all consistent to within a few percent, which casts significant doubt on the reliability of the a priori estimate.

7.3 Modelling

When confronted with significant errors in the observational data that would not calibrate out, a simple 4 Gaussian model for the source was determined with the *AIPS* task, **IMFIT**. The results are the first 4 components in Table 3.8. Even such a simple model was at the limits of what **IMFIT** would do. There is a more general fitting program in **SDE**, **gfit**, which is really just a user friendly interface to an external program, **gaussfit**. It will handle an arbitrary number of components, and allows the initial guesses to be made visually with **SAOimage**, but it can be difficult to get the model to converge, even in simple cases like this one. The VLBI community is probably the most sophisticated group in astronomy when it comes to model-fitting of complicated sources (the Caltech VLBI program **modelfit** is clearly better than any similar program currently found in *AIPS*) but even here the programs are generally fairly difficult to use. I feel that there is a real need for improved modelling programs, both in ease of use and robustness of fit. Model-fitting of simple sources *should* be a routine part of imaging even for connected element interferometers, but until the programs become considerably easier to use, people will not model-fit unless absolutely forced by the project.

As a demonstration of the power of model-fitting to diagnose deconvolution problems, consider Figure 7.2. The left panel is the best uniformly-weighted synthesis image that could be produced with CLEAN, after multiple rounds of self-calibration and subtraction of an 11 mJy offset confusing source some $120''$ to the north-east. This particular CLEAN image has had the residuals reduced to $5 \mu\text{Jy}/\text{beam}$, so the residuals are quite ignorable. The middle panel is an identical CLEAN of the Gaussian model which has been sampled in the image plane and convolved with the dirty beam to be produce a noiseless model dirty map free of aliasing and representation effects. It is true that this source contains a nearly unresolved elliptical Gaussian component that makes it difficult to represent in discretized form — but that is not the problem here. The errors in the middle panel are caused by the CLEAN algorithm, and nothing else. Close examination of the restored images show there to be a nearly one-to-one correspondence in most of the error features, which is quite remarkable given the crudity of the model. The third panel shows the source as is it would be for a perfect reconstruction.

The CLEAN image actually has a peak to far-off-source dynamic range very close to that predicted by the noise estimate from the visibility data. The measured RMS is $61.1 \mu\text{Jy}/\text{beam}$, for a peak-to-noise dynamic range of 24,900. The predicted image plane thermal noise is $62.9 \mu\text{Jy}/\text{beam}$, or 1.985 times the naturally weighted value. The far RMS in the noiseless model data is much better than this, at $14 \mu\text{Jy}/\text{beam}$, though both data and model are producing negative artifacts at the few thousand to one level. As mentioned in Chapter 5, dynamic range really isn't a particularly good measure for deconvolution errors, in that the errors are spatially variable. The numbers quoted in the introduction are real enough, but any measured dynamic range on a CLEAN image which is limited by deconvolution will be a strong function of where one chooses to take the RMS. As shown in

Section 5.2, the worst deconvolution errors will be in the close neighborhood of the source where it is difficult to measure dynamic ranges. Dynamic ranges are occasionally quoted in the case studies, but usually with mention of the most obvious error feature as well. A peak-to-RMS dynamic range is really only meaningful when the worst errors are similar to the RMS. CLEAN is reasonably good at minimizing the far sidelobes of the source, so it is quite possible to have the situation as here, where there are obvious strong deconvolution artifacts in the image, but where the far-off-source RMS is close to the thermal noise.

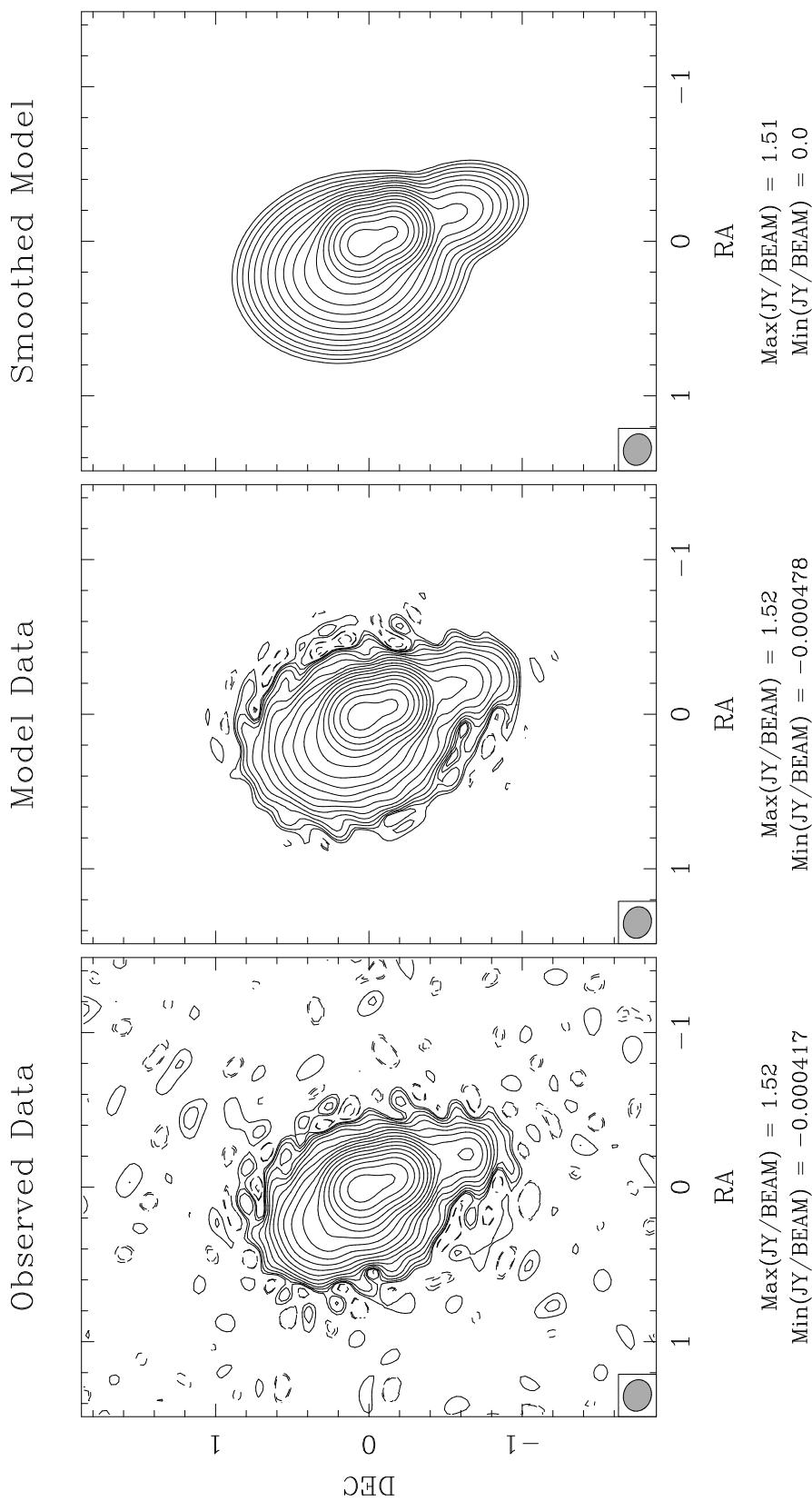


Figure 7.2: Diagnosis of deconvolution problems by modelling. All panels have common contours, with the lowest being 100 μ Jy and four logarithmic contours per decade. The restoring beam is from a fit to the PSF and is .208'' \times .181''.

7.4 X Band Images

Now that we have some idea of what constitutes a “good” deconvolution of the source, Figure 7.3 shows the X band images as a function of the major algorithms and visibility weightings. In particular, notice that the differences between the CLEAN panel of Figure 7.3a and the NNLS panel of Figure 7.3b can all be attributed to improved software introduced in this dissertation. The contours are common to all four pages, with four logarithmic contours per decade. The lowest contour is roughly 3 times the RMS of the least noisy image.

Using a ΔS_{uw} of .1227 Jy, and the thermal noise equations 3.4 and 3.6, the predicted RMS thermal noises for the weightings Uniform, Robust=0, Robust=0.6 and Natural are 62.9, 46.5, 35.1 and 32.1 $\mu\text{Jy}/\text{beam}$, respectively. The measured values for CLEAN are 61.1, 47.9, 38.6 and 40.4 $\mu\text{Jy}/\text{beam}$. For NNLS, the measured values are 60.9, 46.3, 36.6 and 39.1 $\mu\text{Jy}/\text{beam}$. For MEM they are 89.8, 81.1, 74.1 and 107.5 $\mu\text{Jy}/\text{beam}$. For both CLEAN and NNLS, the measured and predicted values track each other fairly well, indicating that the new weighting scheme is performing as claimed in Chapter 3. Notice that the lowest measured RMS is obtained not with natural weighting, but with the Robust=0.6 weighting. This is the value of robustness that best matches the PSF to its fitted Gaussian, and which eliminates the extended shelf so often seen in naturally weighted VLA point spread functions. The difference is not enormous, but it is significant and of course the resolution in the robust beam is somewhat improved over natural weighting.

The CLEAN deconvolutions were done down to a maximum residual of 5 μJy which at a loop gain of 0.1 typically required several hundred thousand components. Bad as are the imaging artifacts here, if one CLEANs down only to the thermal noise instead of well below it, the artifacts become much worse! The MEM deconvolutions were run to 2000 iterations, with a flat default image, $\text{Tflux} = -.1$, and $\text{Sigma} = 4(10)^{-4}$. The MEM solutions at 2000 iterations were noticeably better than at 200 iterations, though the rate of improvement per iteration at that point was extremely slow. Other values of the control parameters were tried, and none made a significant difference beyond that mentioned here. All deconvolutions were loosely boxed to the rough support of the source, into a region determined by eye with 2022 allowed pixels. The NNLS deconvolutions also used this box as the data window.

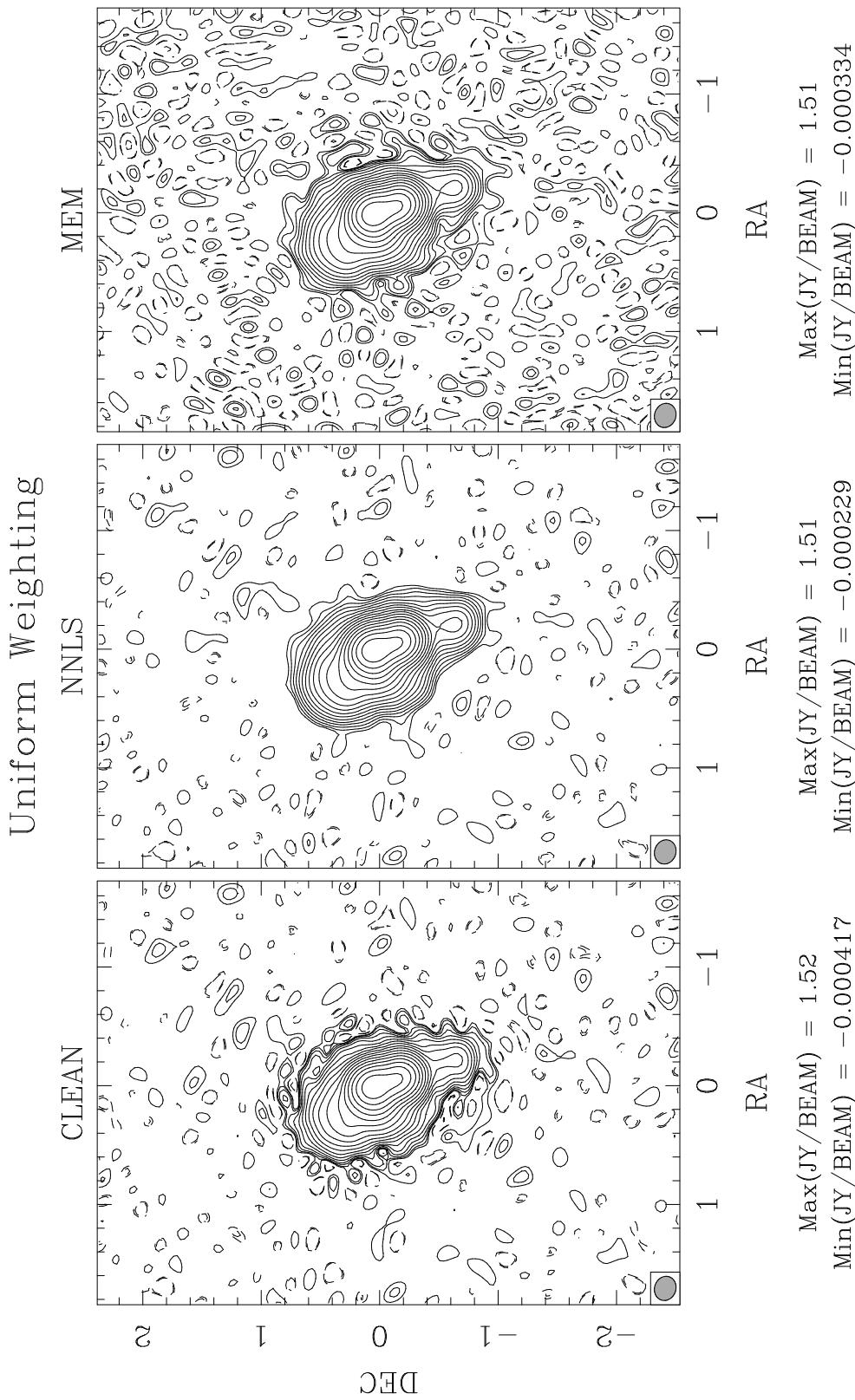


Figure 7.3a: 3C48 X-band images. Lowest contour is 90 $\mu\text{Jy}/\text{beam}$, with four logarithmic contours per decade. Predicted thermal noise is 62.9 $\mu\text{Jy}/\text{beam}$. Measured far-off-source RMS is 61.1, 60.9 and 89.8 $\mu\text{Jy}/\text{beam}$ for CLEAN, NNLS and MEM respectively. The weighting RMS degradation and mean resolution is 1.985 and .194''.

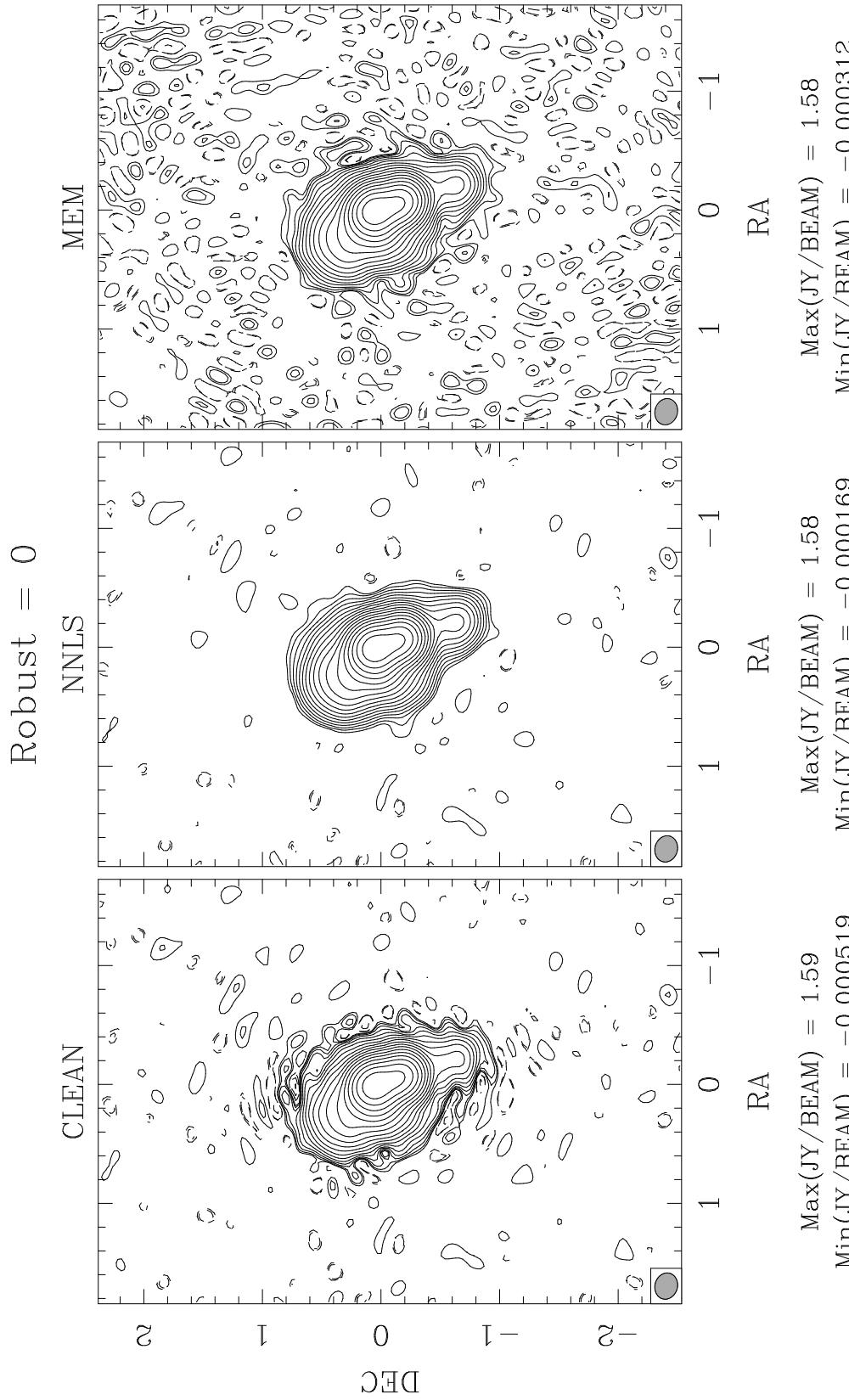


Figure 7.3b: Robustness set to decrease the noise level while maintaining resolution as much as possible. Same contours as in Figure 7.3a. Predicted thermal noise is $46.5 \mu\text{Jy}/\text{beam}$. Measured RMS is $47.9, 46.3$ and $81.1 \mu\text{Jy}/\text{beam}$, left to right. The weighting RMS degradation/resolution is $1.447/207''$.

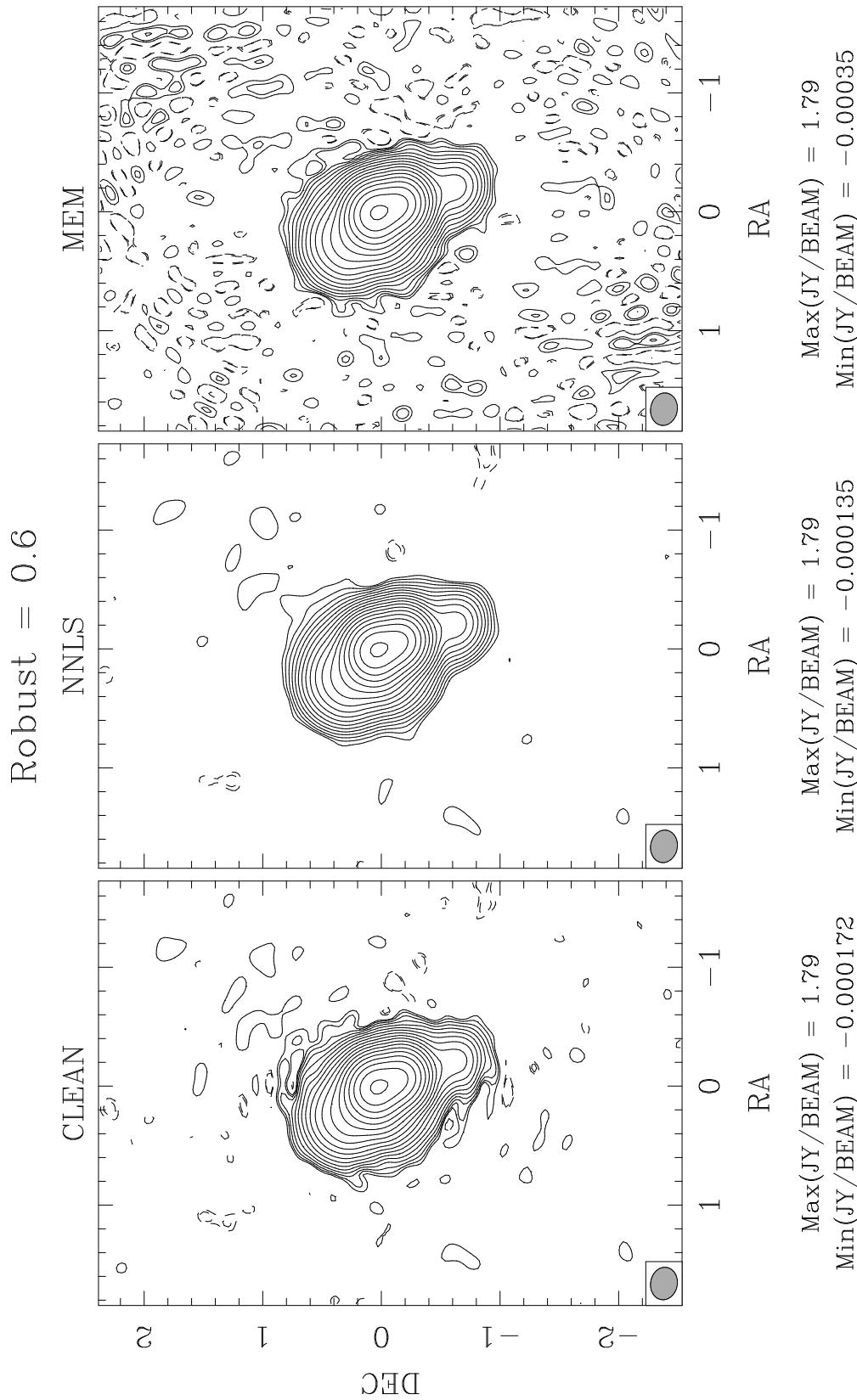


Figure 7.3c: Robustness set to increase resolution over natural weighting, while nearly preserving the thermal RMS and improving the sidelobes. Same contours as before. Predicted thermal noise is $35.1 \mu\text{Jy}/\text{beam}$, and the measured RMS is 38.6, 36.6 and $74.1 \mu\text{Jy}/\text{beam}$. The weighting RMS degradation/resolution is $1.094/.249''$.

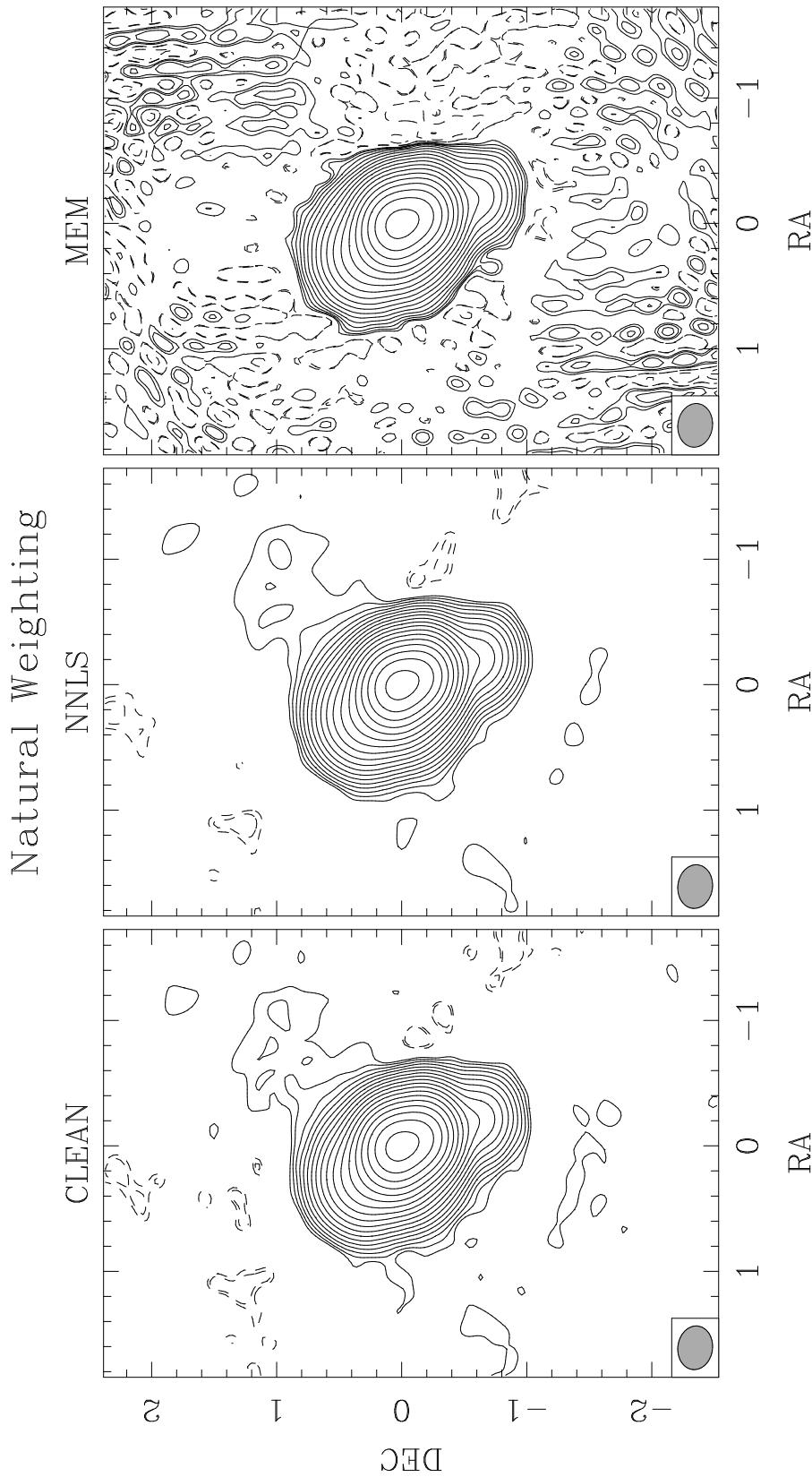


Figure 7.3d: Naturally weighted images. Same contours as before. Predicted thermal noise is $32.1 \mu\text{Jy}/\text{beam}$, and the measured RMS is $40.4, 39.1$ and $107.5 \mu\text{Jy}/\text{beam}$. The weighting RMS degradation is 1.000 by definition, and the mean beamwidth is $.308''$.

7.5 L Band Images

The X band observations were interleaved with L band observations at a frequency of 1.47 GHz. At this frequency and resolution, the source is resolved, but only barely so. The source amplitude plot is given in Figure 7.4, showing both the resolution of the source and the quality of the data. Notice that even at the longest spacings, the lowest visibility is still about 70% of the peak. The source is not resolved by much! Model-fitting to the best image yielded a one component Gaussian model source of $.402'' \times .208'' @ 30.0^\circ$. With the fitted natural beam of $1.891'' \times 1.231'' @ -84.0^\circ$, this means the source width varies with position angle from extremes of $.169$ to $.213$ of the fitted beam. While small, this is still quite large enough for CLEAN to have trouble, as shown in Figure 5.9. No algorithmic comparison was done on this data, as it would have been largely redundant with the DA193 case study in Chapter 9.

The L band images were designed for maximum sensitivity, and so natural weighting was used throughout. The source was thought sufficiently compact that NNLS would not have difficulty removing the extended wings of the beam, and intermediate values of the robustness were not explored. In retrospect, a robustly weighted beam would probably have been a good idea, especially given that the image is still limited well above the thermal noise determined by internal visibility scatter. The problem might be the wings of the natural beam, or we might be approaching the hardware limits of the instrument. More work should probably be done on this data, and the image in Figure 7.5 is regarded as preliminary.

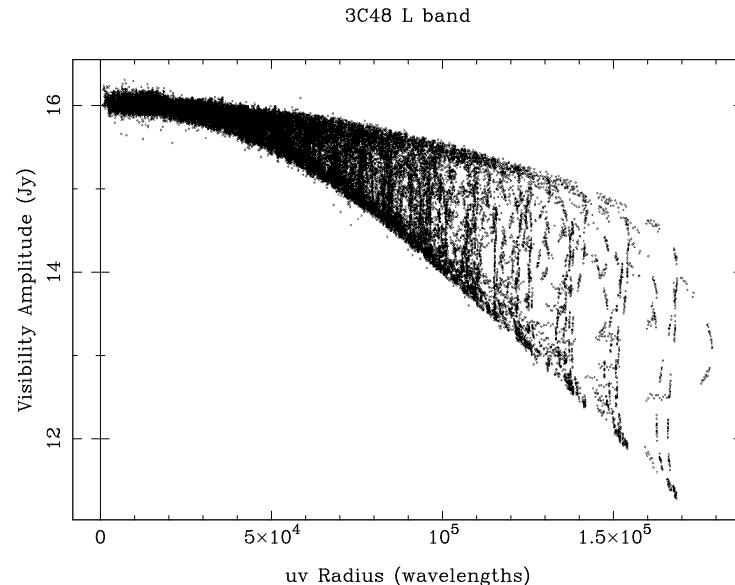


Figure 7.4: Visibility amplitude plot. The visibilities have been time averaged down to 2 minutes in this plot, and all points are shown. Notice the numerical scale. The source is clearly resolved, but only slightly.

After calibration, flagging, and several cycles of self-calibration within AIPS by Craig Walker, the data were exported to SDE for subsequent processing. Since the NNLS tasks currently lack multiple field capability, the data were first imaged a final time with MX and lightly CLEANed. Of the 4000 total components, 150 were removed from the offset field, yielding 59.6 mJy CLEANed flux. This was subtracted from the visibility database with UVSUB and the data exported to SDE. Several additional cycles of self-calibration and NNLS imaging of 3C48 were performed, with results near the source essentially as good as shown in Figure 7.5b. The images were made with cellsize of $.16''$, which greatly oversamples the beam, and the images at this stage were 256 pixels on a side. Eventually, however, we decided to image a large field in hopes of detecting extended diffuse emission at significant distances from the main source. The NNLS model components were padded with zeroes out to 2048^2 . The offset source CLEAN components were added back into the component model and the resulting composite used to produce a 2048^2 restored image from a 4096^2 dirty map. While perhaps not surprising in hindsight, the very slight calibration improvements in the visibility data from the NNLS iterations were not reflected in the MX CLEAN components on the offset source, and a calibration error pattern from a 60 mJy source was visible over more than half of the restored image! When working at this level, calibration changes *must* be done in a consistent manner. The gain corrections from the best offset subtracted visibility database were transferred to the original database prior to subtraction, and a final conventional NNLS imaging sequence was performed, working completely with these huge images. This proved to be possible, though very close to the limit of the workstation's virtual memory. SDE operates under the assumption that virtual memory is plentiful and that all the intermediate images necessary to the deconvolution can be held in memory at once. In situations like this, the assumption begins to break down. Several normally automated operations had to be performed "by hand" with separate tasks, but the project was barely possible. It need hardly be mentioned that the NNLS deconvolution windows are as tiny as usual, but that they are simply placed on an enormous otherwise-empty region of sky. Once the deconvolution model is determined by solving for the flux in these tiny windows, the restored image can be made as usual over the entire area. NNLS deconvolution certainly needs to be incorporated into a task with knowledge of multiple fields, to avoid the necessity for heroic measures like these. Situations like this might also benefit from calculation of the relatively modest number of pixels in the deconvolution window by a direct rather than a gridded transform. Even the beam elements connecting the pixels in the deconvolution windows might be calculated in an ungridded manner, though this latter idea is probably still computationally infeasible at the moment.

Once the data were processed in a consistent manner, the third offset source to the south was detected. It is only about 1 mJy, but still a 13σ detection! The image presented here placed a small flux window around it, and allowed the deconvolution algorithm to process it properly. When the small source entered the restored image through the residuals, the peak flux was 1.073 mJy/beam, and when it was deconvolved properly it dropped to 1.068 mJy/beam. While mildly surprising that the peak flux drops rather than rises after

deconvolution, clearly the photometry of faint sources is not greatly dependent on whether they enter the image through components or residuals. The source is real, and was also detected at X band.

The flux and data windows used in the final deconvolution were a circle of radius 10 pixels, centered on the southern small source, a circle of radius 15 pixels, centered on the northern offset source, a circle of radius 33 pixels, centered on 3C48; the remaining several thousand pixels out of a total 5500 were placed in an oblong shape out towards the north-east from 3C48, where earlier images had hinted there might be emission. The resulting full image is given Figure 7.5a, and expanded versions of the interesting subfields in Figure 7.5b. The predicted thermal noise based on the scatter in the data is $54.0 \mu\text{Jy}$, while the measured off-source RMS is $91.0 \mu\text{Jy}$, so clearly there is something limiting the deconvolution beyond the thermal noise.

3C48 L band

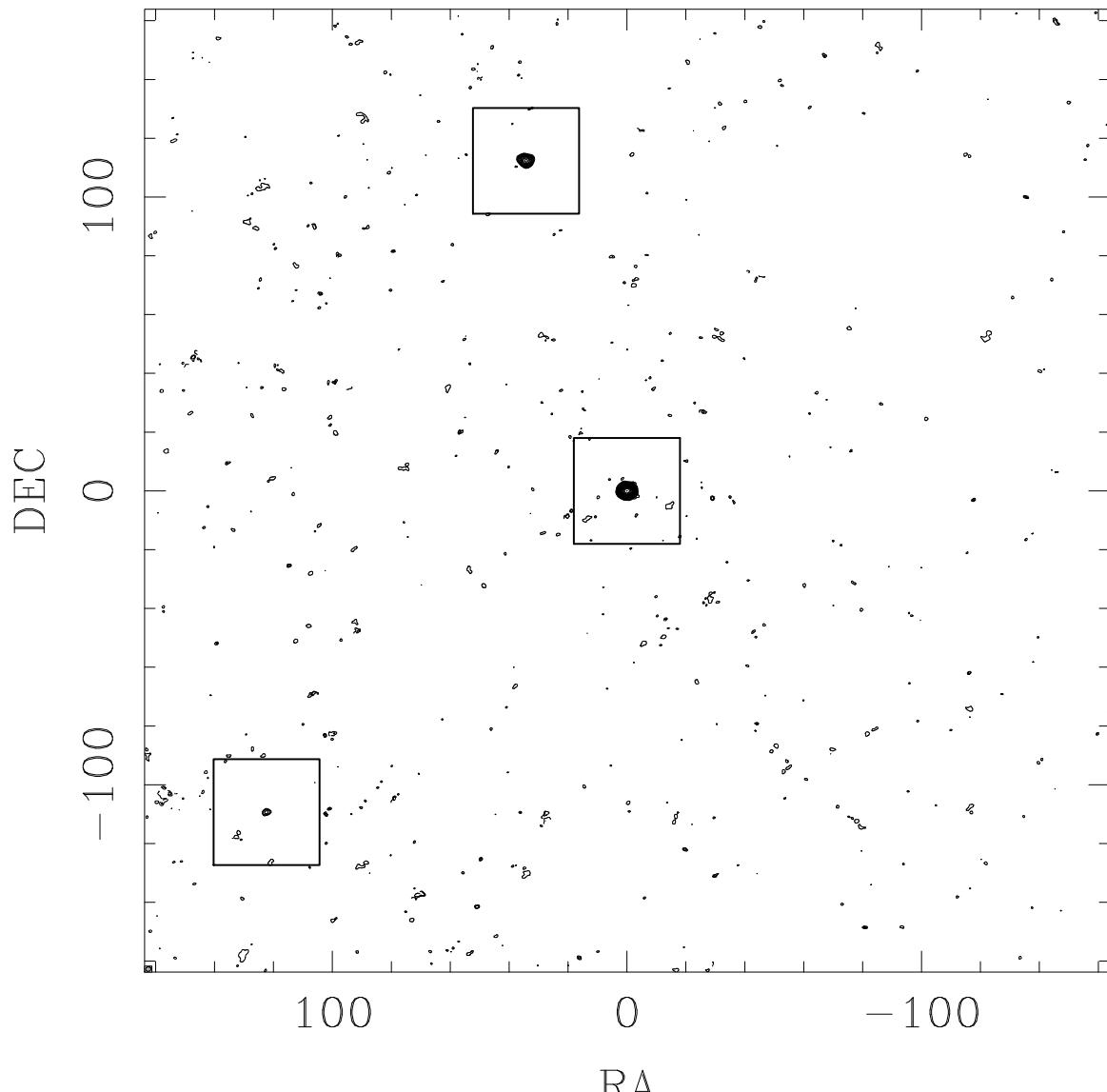


Figure 7.5a: Wide field L band naturally weighted image. The dynamic range is 180,000:1, with no obvious artifacts significantly above the measured RMS. The peak emission in the Northern offset source is 58 mJy/beam and for the southern source, 1 mJy. The lowest contour is 3 times the off-source RMS, or 273 μ Jy/beam, and there are four logarithmic contours per decade. The restoring beamwidth is $1.891'' \times 1.231''$ and is shown in the lower left corner.

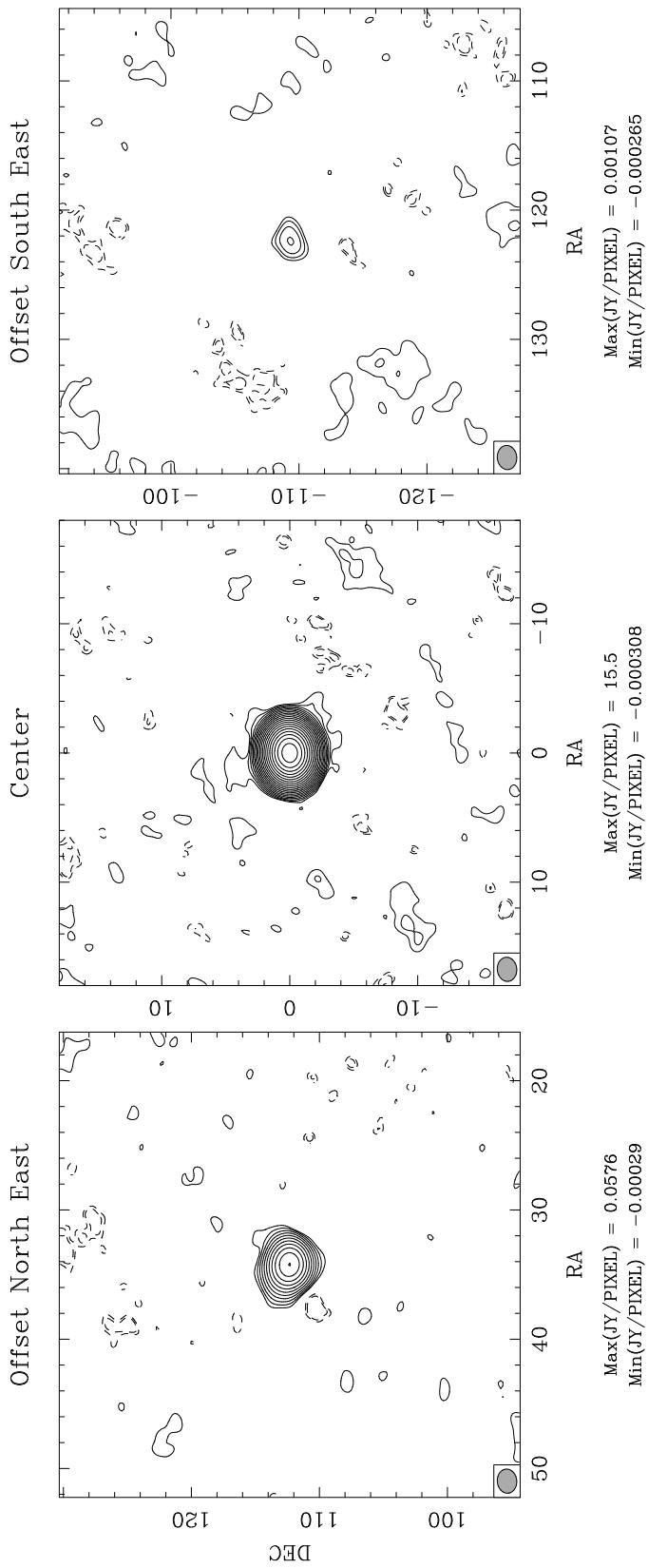


Figure 7.5b: Expanded sub-images from Figure 7.5a. Lowest contour is 2 times the offsource RMS in the full image. Four logarithmic contours per decade.

7.6 Discussion

The astrophysics of 3C48 is discussed extensively in Wilkinson et. al (1990 and 1991). The germane points here are that the source is extremely small for a radio loud quasar on both linear and angular scales. It is believed that the central engine is embedded in a dense clump of gas in the interstellar medium of the host galaxy, which is confining the jet to an abnormal size. An obvious question is whether the jet has been limited in size over its entire history, or whether the collision with the dense gas is a comparatively recent event. The objective of the L band observation was to see if the jet had ever punched all the way through the confining gas in the past. If so, there would be some hope of detecting emission from the older ejected material, which might be at a considerable distance from the main source. The answer appears to be that if such material is there, it is not detected in this observation. The northern offset source is at the position angle of the jet, of course, but one would expect some sort of diffuse emission if they were related. There is nothing linking the two otherwise distinct compact sources.

The morphology of the 3C48 jet is extremely unusual. There is a distinct change in the properties of extended extragalactic radio sources at an absolute luminosity of about $P(14 \text{ GHz}) \sim 10^{25} \text{ W/Hz}$, which divides the sources into the Fanaroff and Riley classes I and II, (Fanaroff and Riley, 1974). By power, 3C48 is well into the FR II category, where one would expect a supersonic jet, narrow in angle and fairly straight. In this source, the turbulent-looking wide-angle jet to the north seen on VLBI size scales resembles the subsonic jet in a far less powerful FR I radio source – or at least half of one. Since turbulent flow involves material travelling at many different angles to the line of sight, it is difficult to hide such a jet by invoking beaming arguments and nearly all FR I sources are two sided. Prior to these observations, no counterjet had been seen in 3C48, and the source had been considered intrinsically one sided. The small extension to the south in the X band images is the first detection of the counterjet in this source. Without invoking beaming, it is difficult to imagine ways of producing such a powerful asymmetry, to say nothing of a completely one-sided jet. At least now we have a direct measurement of the degree of asymmetry. Slices through the source perpendicular to the jet may yield information about the strength of the confining process, and it is here that the high quality deconvolution of the X band images may prove most scientifically useful. A source surrounded by a spurious negative will pass through zero in a slice much earlier than otherwise, and the measured slope and sharpness at the edge will be higher than actually true. The interpretation of such detail is a matter for specialists, but now the observational images are of sufficient quality to warrant the scrutiny.

Chapter 8

Case Study: SN1987A

The work in this chapter resulted from a 5 month visit to the Australia Telescope National Facility, kindly supported by the Australian Commonwealth Scientific Industrial Research Organization (CSIRO). Much of this chapter has been published as Staveley-Smith et al. (1993a) and Briggs (1994).

Supernova 1987A in the Large Magellanic Cloud presents an unprecedented opportunity to observe the detailed evolution of a supernova at all wavelengths. While optical observations with the HST Faint Object Camera (FOC) have obtained resolutions of $.1''$, we are limited in the radio to the resolution obtainable with the Australia Telescope Compact Array. At the highest frequency of 8.8 GHz, this corresponds to a synthesized beam width of $.9''$. At this resolution the radio supernova is distinct from a point source, but few physical conclusions can be drawn. We present here super-resolved images from the ATCA with an effective resolution of $.4''$ – $.55''$. These reveal a spherical shell-like structure with a radius of $.6''$, and an additional component of emission aligned with the circumstellar ring imaged with the FOC. Care is required due to the delicate nature of the imaging problem, and we present evidence that the overall image structure is plausible.

8.1 Introduction

The Australia Telescope is the premier imaging radio interferometer in the southern hemisphere, and the only modern instrument able to observe SN1987A. The compact array consists of 6 22m antennas with a maximum baseline length of 6 kilometers. Additional Australian antennas can be combined with the ATCA for Very Long Baseline observations, but to date there are no detections of the supernova on baselines longer than 6 km.

Deconvolution in radio interferometry is more generally used to remove the far-off sidelobes caused by the incomplete sampling of the Fourier plane than for super resolution. Even though it has been known for many years that moderate super resolution can be successful on simple sources,¹ in practice it is often the case that desired higher resolution can be had directly by using data from a different instrument, different array configuration or

¹“a modest degree of superresolution (not more than a factor of two) will often be possible and need not be treated with suspicion.” (Narayan and Nityananda, 1986, p. 134)

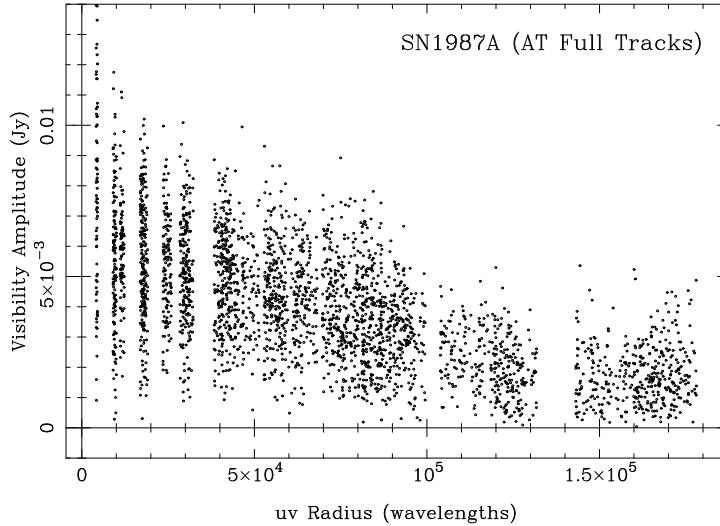


Figure 8.1: Visibility amplitude plot. The visibilities have been time averaged down to 10 minutes in this plot, and all points with full weight are shown.

different observing frequency. SN1987A is unusual in that it is a crucially important object where higher resolution radio imaging data is simply not available. While the supernova will expand with time and thus provide a better look at the remnant to come, the existing data are all that will ever be obtained of the supernova at the current epoch of evolution. There is great incentive to extract as much spatial information as possible from the existing data. Even a map of very low dynamic range will be extremely valuable astrophysically, so long as the features interpreted are reliable.

8.2 The Data

The data used for these images were taken by Lister Staveley-Smith on October 21, 1992 and January 4–5, 1993. The source was followed for a full 12 hour track each day, with a small amount of data rejected in periods of poor atmospheric phase stability. After external calibration by him, 116396 good visibility record at 15 second averaging time were retained. No self-calibration was performed. The total bandwidth was 128 MHz in 32×4 MHz channels, centered on 8.8 GHz. The typical system temperature was 65 K, though the outrigger antenna which participates in the longest baselines had recently been upgraded with an improved receiver and had a system temperature half that of the others.

The radial visibility amplitude plot in Figure 8.1 shows that the source is indeed resolved on these baselines, and that there does appear to be compact structure on the longest baselines. The u - v coverage was used as a test case for the new weighting routines, and is given back in Figure 3.12b.

8.3 Deconvolution

The dirty maps were made with the conventional gridded uniform weighting. The cell size $0.1''$ was selected as adequate to oversample the maximum effective resolution, and most dirty maps were of size 512^2 . This is exactly the case explored in Chapter 3, so the discussion of Section 3.7.6 applies here as well, along with the uniformly weighted beam presented in Figure 3.19. The dependence of the final maps on cell size was very weak, so long as the PSF was highly oversampled. Iterative algorithms were started with a flat default, as more sophisticated choices of default were found to make little difference. The averaging time was found to make a significant difference in the final resolution achieved, presumably for the reasons discussed in Section 3.8.3, and the data were not averaged beyond the initial 15 second records. Several weighting schemes were explored, and the conventional uniform weighting selected as being the most useful. This topic is covered in more detail in Section 8.5.

The weights of the visibility data as they come off the ATCA are simply constant and did not reflect the significant differences in system temperatures present. The weights were multiplied by 2 for all baselines which used the upgraded outrigger antenna; all the deconvolution programs used treat the weights statistically. Surprisingly, this resulted in very little difference in the final maps, when compared with maps prepared from data where all baselines were weighted equally.

The deconvolution algorithms Maximum Entropy, CLEAN, Non-Negative Least Squares, Maximum Emptiness, Gerchberg-Saxton-Papoulis, Richardson-Lucy and Singular Value Decomposition were used for the initial mapping. (Descriptions of these algorithms are given in Section 2.6.) All algorithms produced nearly identical results at the nominal fitted restoring beamwidth of $.98'' \times .83''$, though there were considerable differences in the robustness to support information, super-resolution properties and convergence speed. Several examples of the images at nominal resolution are given in Figure 8.2.

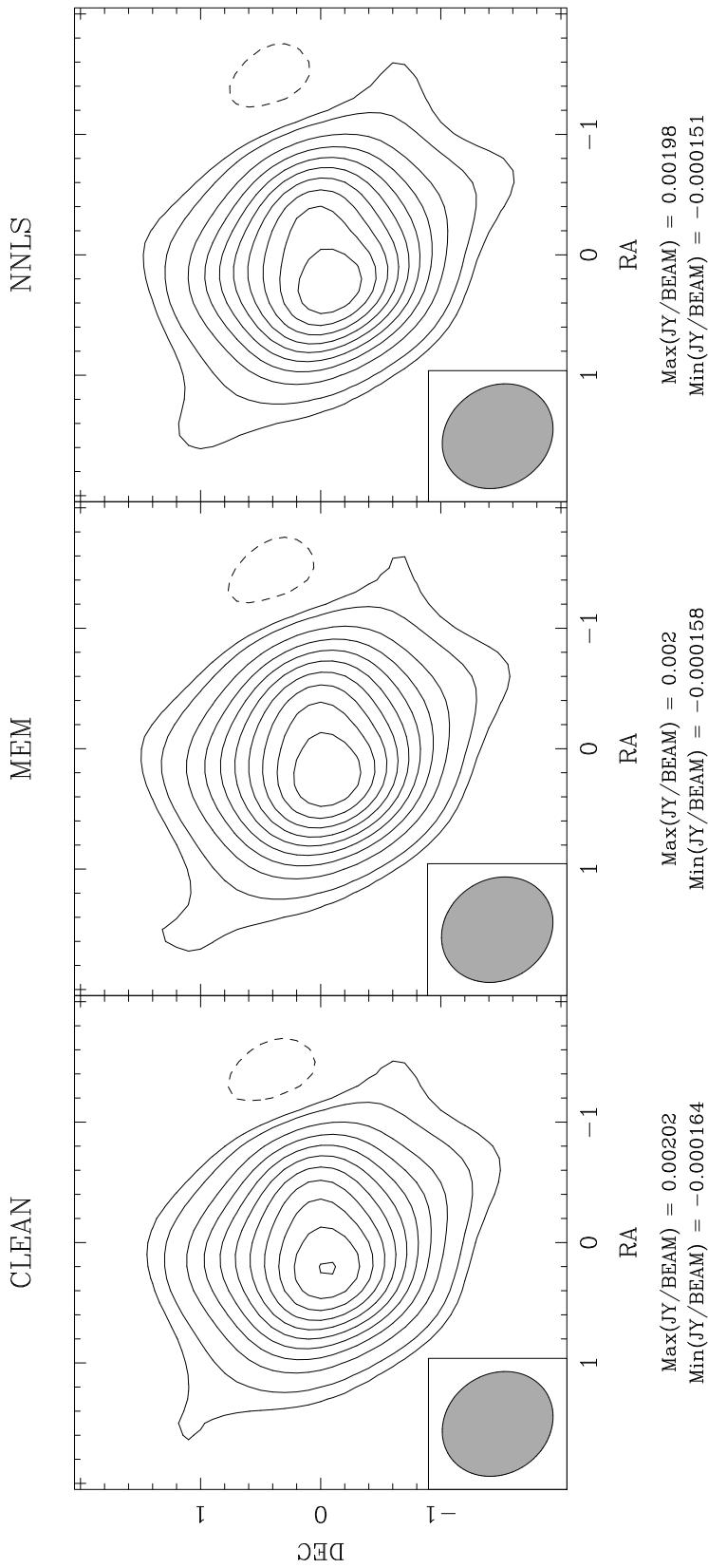


Figure 8.2: Nominal uniformly weighted images for different algorithms. There is no significant difference between these images or the nominal images from the other deconvolution algorithms. Contours are linear at intervals of 10% of the MEM peak, with an additional contour at 5%. All contours levels in this chapter are reflected symmetrically around zero. The shaded ellipse is the fitted beamwidth of $.98'' \times .83''$.

All of the algorithms used here operate in the image plane, and all use a component model of the sky which is measured in Jy/pixel. This model is then convolved with a restoring beam representing the desired final resolution, and the residuals are added back to produce the final image normally interpreted by the astronomer. For the purposes of the super resolution study, it is often more productive to examine the sky model directly, though care should be taken to view features present in the components with proper skepticism. It is in the nature of the high resolution sky models that the differences between algorithms are apparent. In particular, one of the most fundamental distinctions is that some of these algorithms incorporate a smoothness constraint, either explicitly or implicitly as a result of finite iteration. The result is that these component models have a finite (possibly space variant) “maximum resolution” that is roughly independent of the grid spacing. Two of these algorithms, CLEAN and NNLS, have no such constraint and will generate features at whatever resolution is requested of them. These sky models must generally be smoothed down before much sensible interpretation can be done. The following general behavior was noted in the context of this project.

Maximum Entropy This algorithm produced what appears to be the best reconstruction found. It agrees with the other algorithms on the major features, produces a plausible physical image, performs well on simulated data with a high degree of super resolution, converges smoothly to a reasonable answer, and is robust to support constraints. For this project it is the algorithm of choice. The only disadvantage is that an extreme number of iterations is required for the highest resolution.

Non-Negative Least Squares NNLS is almost certainly performing better than MEM at the nominal resolution, yet the limiting factor for both is the thermal and confusion noise and the images are equivalent. The important question is at exactly what resolution the NNLS solution breaks down beyond that of the MEM solution. This is not precisely known, and the prudent investigator should examine both algorithms at a variety of restoring resolutions.

CLEAN While quite acceptable at the nominal resolution, CLEAN is *not* recommended for super-resolution purposes. The component image shares the same speckled appearance as produced by the NNLS algorithm, but it produces spurious structure much sooner in simulated tests. An experimental positivity constrained version of CLEAN was examined, and it fared no better.

Maximum Emptiness This algorithm minimizes the L1 norm of the component solution while satisfying the convolution equation within the noise. It was inspired by a paper which suggests that CLEAN is approximately minimizing this criterion. In fact, when terminated at a reasonable number of iterations, it behaves more similarly to MEM. The emptiness regularization appears somewhat less powerful than the entropy — it converges slowly and the degree of super resolution is less than with MEM.

Gerchberg-Saxon-Papoulis This algorithm merely forces agreement with the data in the Fourier plane, and enforces positivity and support in the image plane. As such, it

is very simple conceptually, and encouraging when it agrees with other methods. It produced a final result very similar to that of MEM, but diverged without support constraint.

Richardson-Lucy This algorithm was comparable to Maximum Emptiness in that it produced a physically reasonable model but at a lower resolution than MEM. This algorithm is highly favored in the optical community, but less suited to the needs of interferometry in that it requires a nonnegative PSF; the ad hoc modification of clipping the dirty map and PSF to nonnegative values is crude. The theoretical formalism is quite different from the other algorithms, and again it is encouraging that it works here as well as it does.

Singular Value Decomposition Direct inversion of the deconvolution equation can also be regularized with a Singular Value decomposition of the beam matrix. With heuristic selection of an appropriate cutoff in singular value, a model was obtained that resembled the MEM model, but which still contained negative flux at an unacceptably high level.

As to the specific values of the control parameters, the MEM deconvolutions used a flat default image, $Tflux = -.02$, $Sigma = 3(10)^{-5}$, and 2000 iterations unless described otherwise. The Maximum Emptiness deconvolutions used the same control parameters, and images are given with 40 and 2000 iterations. The CLEAN deconvolution used 500 components at .10 loop gain. GSP used an acceleration parameter of 0 and 5000 iterations. The SVD deconvolutions used both 33 and 50 singular values. All of the windowed deconvolutions used the support window shown in Figure 8.4. NNLS used this as the flux window, and a circle of radius 30 pixels, centered on the supernova, as the data window.

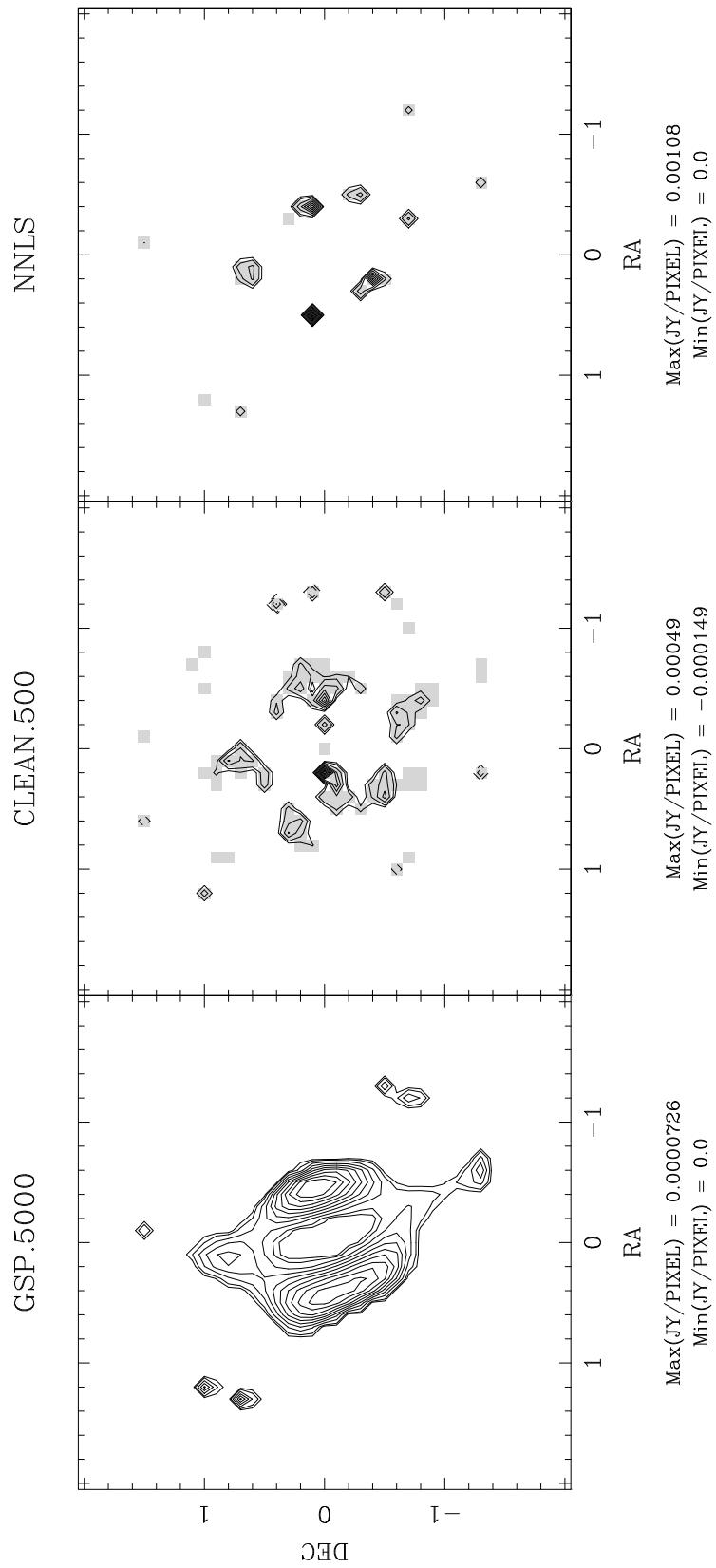


Figure 8.3a: Sky models for different deconvolution algorithms. The CLEAN and MEM panels have a light grey shading superimposed wherever the component model is nonzero. Numbers in the panel title are the number of algorithm iterations. Contours are 5, 10, 20, 30%... of the peak in each panel.

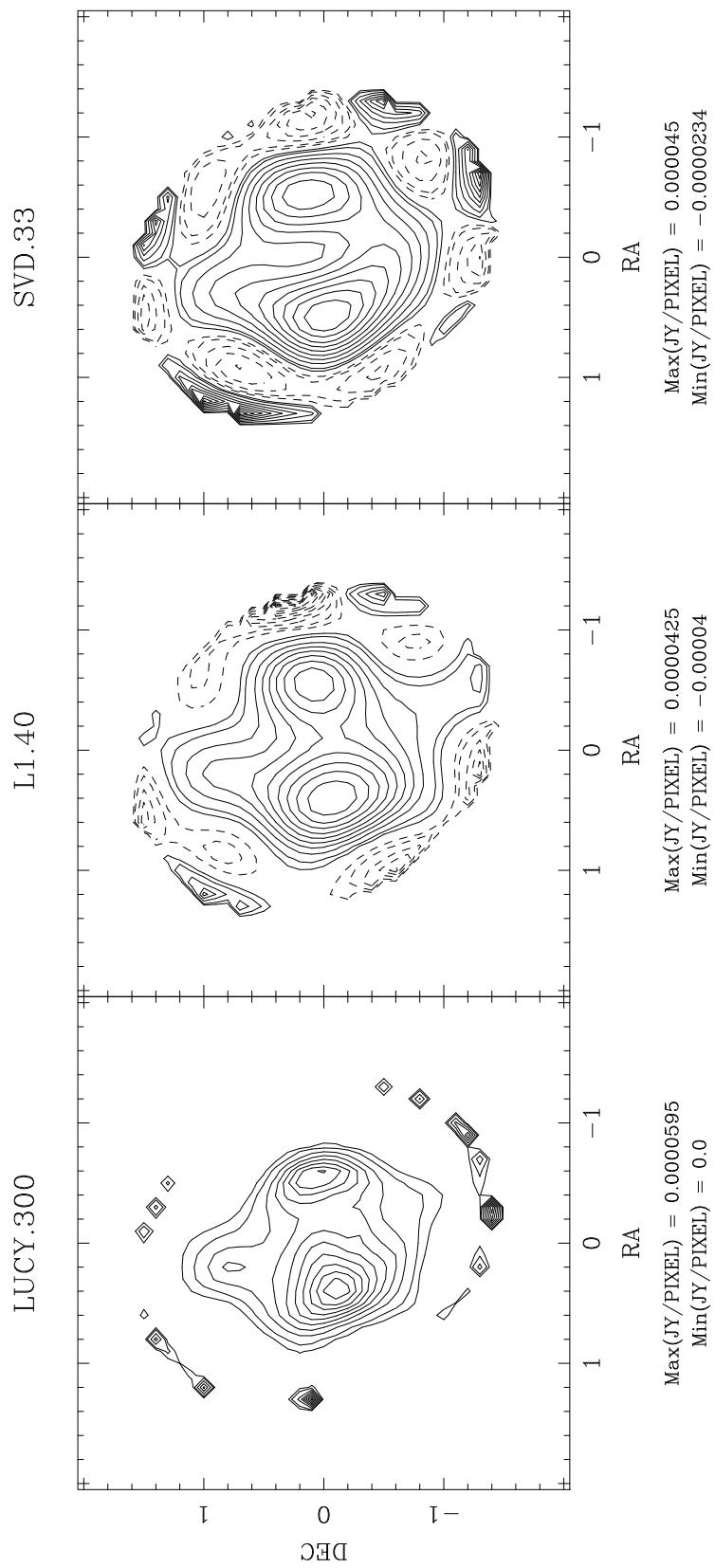


Figure 8.3b: More sky models for different deconvolution algorithms. The “iteration number” in the SVD panel is the number of singular vectors used in the solution. Same percentage contours as before.

8.3.1 Support Constraints

The support constraints ranged from slightly to extremely important, depending on the algorithm. GSP actually diverged, unless a tight support constraint was used to confine the emission to the immediate region of the supernova. SVD deconvolution also requires the support constraint for fundamental reasons — without it, the algorithm simply returns the dirty map. NNLS requires reasonable support information for more pragmatic reasons; the problem becomes intractably large without it. MEM, CLEAN, L1, & Richardson-Lucy all produced reasonable answers with no support information, but all did somewhat better when it was included. The price one pays for the better deconvolution on source is that inevitably some of the noise power is transferred into the component model. If a hard support window is used, the interaction of the noise power in the components and the edge of the window can produce unsightly isolated artifacts. Since this was a fairly minor effect with MEM, the extra resolution was considered worth the artifacts. The extra information provided to the algorithm is physically reasonable, after all. The more robust algorithms were used to verify that there was no emission outside the window that could not be accounted for by the noise. Figure 8.4 shows the MEM model image after deconvolution with and without a support window. Notice that the ratio of the peak to central depression has increased considerably. Other than this example, all deconvolutions in this chapter are done with a hard support window.

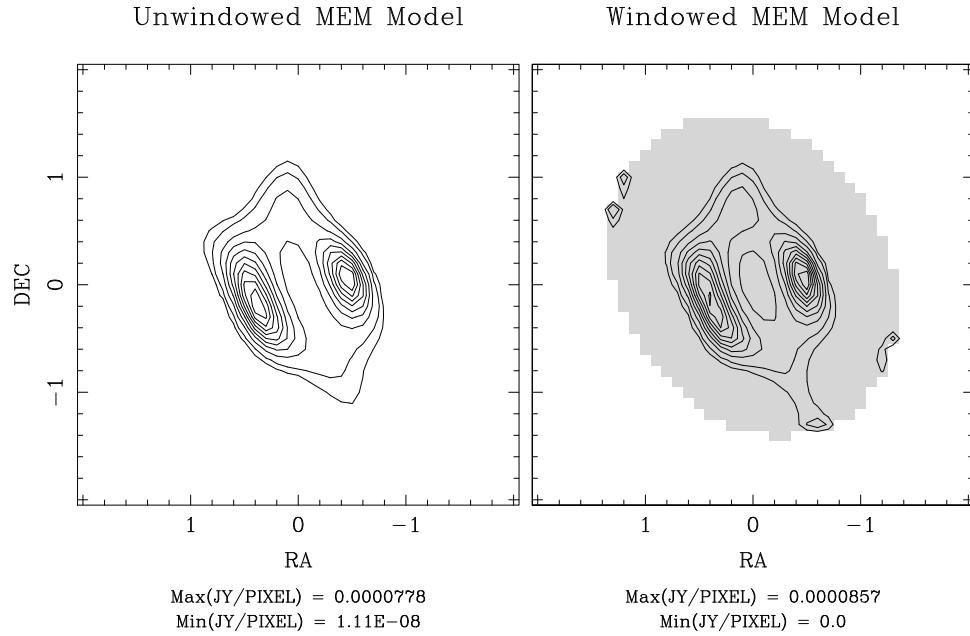


Figure 8.4: MEM sky models with and without a support window. The shaded region in the right panel is the support window where the algorithm has been allowed to place flux. The ratio of peak to central depression has gone from 7.7:1 to 17.2:1 by including the support information. Contours are 5, 10, 20, 30% ... of the peak in each panel.

8.3.2 Cross Validation

An obvious method of trying to decide if image features are believable is to observe more than once and see if they recur. Or more likely in practice, one can divide the data into independent subsets and image each individually. In this case, we actually had two completely distinct observations, where the weather and at least some of the instrumental systematics were completely distinct. The evolution of the supernova itself was not expected to be particularly significant on the timescale of three months. While the images of the data subsets are appreciably noisier than the composite set, the most essential features of the shell like structure, central depression and hotspots are present in both epochs.

An effort was made to determine if it was systematic differences due to the two observation dates that was driving the image differences, or simply random fluctuations due to the independence of the data. Homogenous but independent subsets of the full data set were created by randomly assigning each visibility to one subset or the other. The resulting data sets had nearly identical PSFs and common weather effects, but were statistically independent. Typical runs of this procedure produced variations between the subsets of the same order as shown here, leading to the conclusion that we are seeing primarily statistical fluctuation.

The ill-conditioned nature of the super-resolution problem means that features much larger than the thermal noise can vary when the resolution is pushed beyond the nominal.

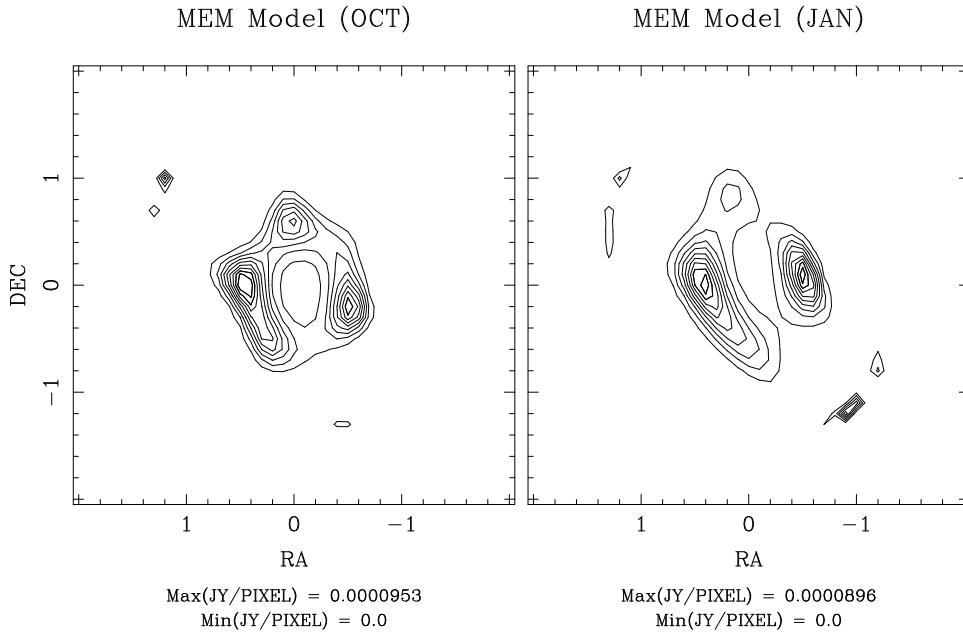


Figure 8.5: Cross validation: maximum entropy models for two independent epochs.

8.4 Algorithmic Convergence

It is commonly held that MEM algorithms require a relatively modest number of iterations to converge. (“All practical MEM algorithms are found to require on the order of 30 to 50 iterations to converge. This convergence behavior is not now understood in any fundamental sense.” (Press et al., 1986, p.816)) Prevailing opinion seems to be that it is at best useless, and possibly dangerous, to iteration beyond this range. In this application, “excessive” iteration seems actually to be beneficial. While the deconvolution has converged at nominal and mildly super-resolved resolutions by 50 iterations, the MEM model continues to sharpen well beyond this point. A strength of the algorithm when used with a flat default is that the model is only sharpened to increase agreement with the data within the noise. In that sense, the high spatial frequency structure produced can be taken as a lower limit to that actually present. There is a danger that forcing agreement to noisy data better than warranted may create spurious peaks, but in this case the parameter controlling required agreement to the data has been varied considerably with relatively little impact on the restored image. The sharpening peaks do seem to be mandated by the data. The extremely long convergence time demonstrated by Figure 8.6 may actually be a reflection of limitations in the simple Cornwell-Evans MEM algorithm, or might merely be indicative of the demanding and somewhat unusual requirements of this application. It would be interesting to repeat these tests with one of the more sophisticated MEMSYS algorithms, but this has not been done. Certainly it does not appear detrimental to run the MEM implementation `vm` to extreme numbers of iterations.

The property of converging smoothly in the limit of extreme iteration is definitely not shared by all algorithms examined. Figure 8.7 shows models from algorithms less forgiving in this respect. The Richardson-Lucy algorithm is actually doing reasonably well. It’s sharpened up the image somewhat at the cost of speckling the model and increased artifacts at the edge and center, but is otherwise still fairly reasonable. The L1 algorithm has completely ceased to resemble a physical model, however. Perhaps more interesting is that the solution has come to resemble a smoothed version of the CLEAN model. Both L1 and CLEAN have placed the eastern hotspot at the peak of the dirty map — well inside the shell structure revealed by the other more physically plausible models. Both have nearly destroyed the central depression and shell. Similarly, the difference between using 33 versus 50 singular vectors in the SVD deconvolution versus 50 is quite dramatic, and over-‘iteration’ completely destroys the physical structure in the model. Unfortunately, while the residual image gives some clue as to the proper number of iterations, when operating in the super-resolution regime, there is no substitute for directly examining the model and terminating the iteration by eye. It is best to choose an algorithm which degrades gracefully if the wrong number of iterations is selected.

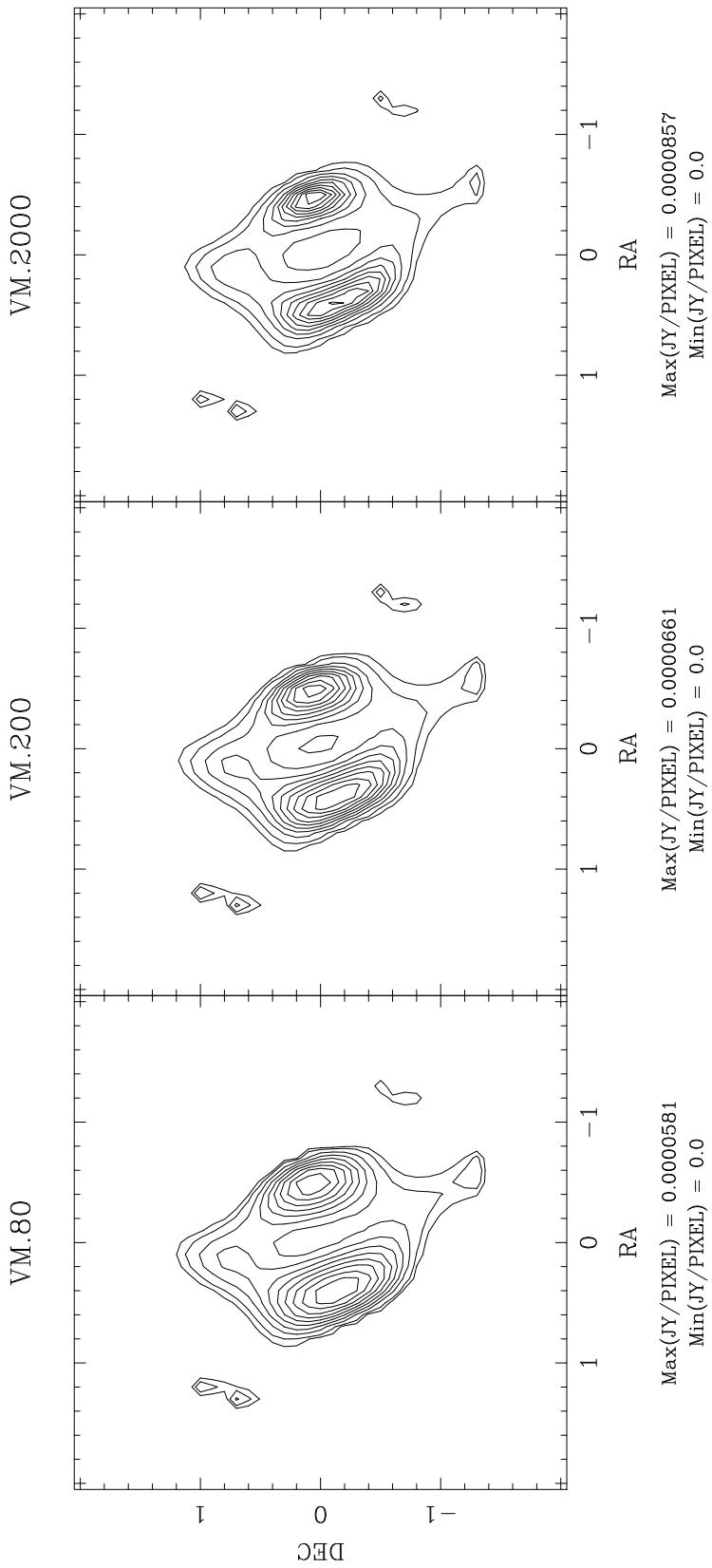


Figure 8.6: Convergence: maximum entropy models as a function of iteration. From the left, 80, 200, and 2000 iterations. Notice that the peak model value in Jy/pixel is rising as the hotspots sharpen. The central depression is also decreasing in absolute units. The ratio of peak to minimum value in the central depression is 7.8, 11.6, and 17.2 respectively. Contours are 5, 10, 20, 30%... of the peak in each panel.

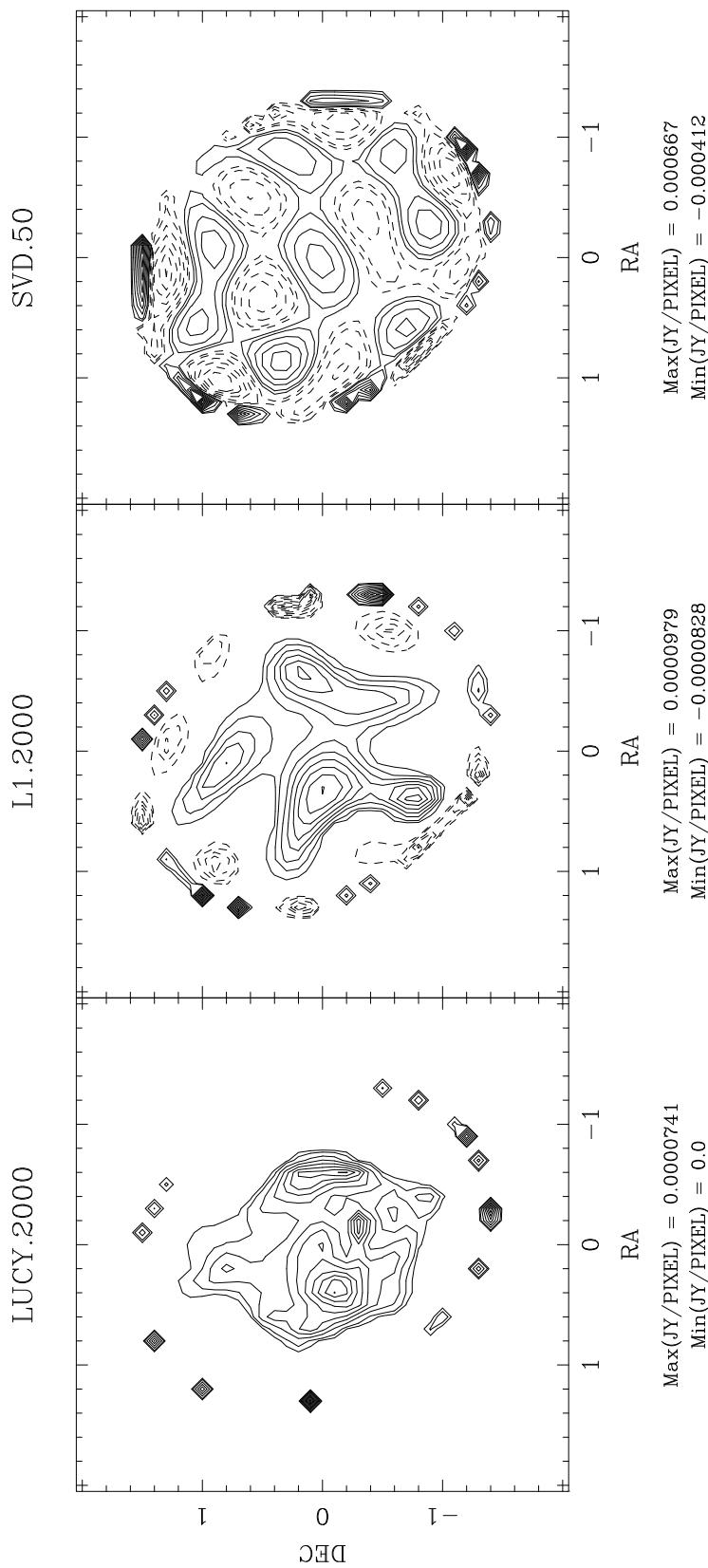


Figure 8.7: Over-iteration: sky models where the algorithm has been iterated past the point of optimum restoration. These models still recover essentially the same image as all the others, at the nominal resolution. Contours are 5, 10, 20, 30%... of the peak in each panel.

8.4.1 Linear Summation

A flaw with imaging subsets of the data is that each subset is appreciably noisier than the whole. One is never sure what can or cannot be attributed to the fluctuations of thermal noise. Another approach is to image subsets independently, and then combine the resulting models in a linear sum. Deconvolution of interferometer data is an intrinsically nonlinear operation, and it is the nonlinear operations in the algorithms that make it possible at all. In most cases, it is a much better idea to deconvolve an entire data set at once than to deconvolve separately and later linearly combine the resulting images. The data in the different subsets do not get a chance to combine nonlinearly in the deconvolution, and the result is usually an inferior image. One can reverse the process and do this deliberately. If the linear sum image is close to the image from deconvolving all the data globally, one can have confidence that it is not some peculiarity in the nonlinear operator of the deconvolution algorithm that is driving the image features. This technique is demonstrated in Figure 8.8, and the reconstruction is quite robust to the procedure. In some crowded-field situations, the changing PSF due to the different $u-v$ coverage of the data subsets will cause the deconvolution errors to be distributed differently in each subset image, and deconvolution errors far from the source will average down roughly independently. In such a situation, averaging independent snapshot deconvolutions may help more than it hurts.

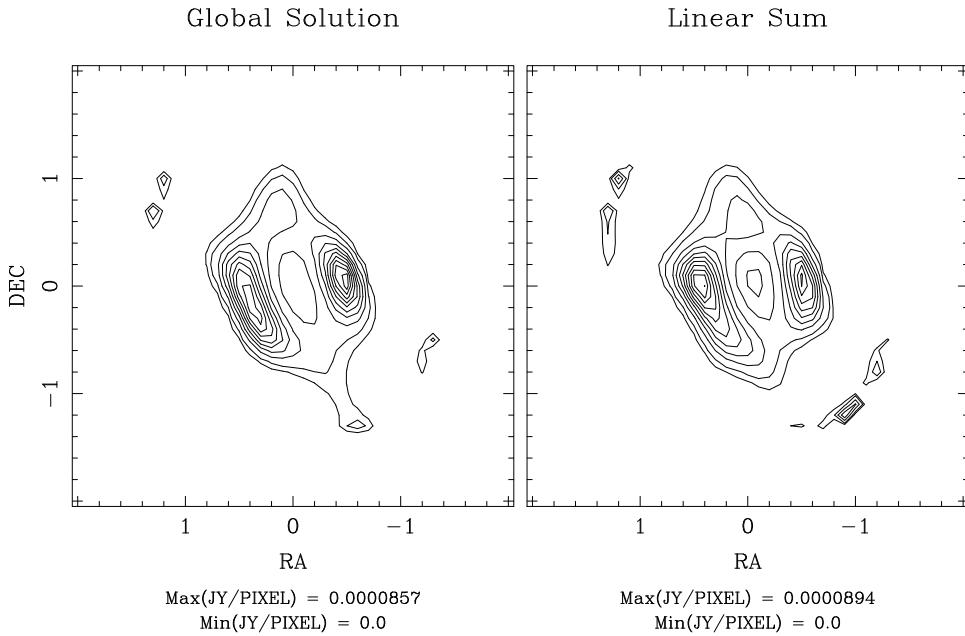


Figure 8.8: Linear combination of subset MEM models: the OCT and JAN subset models from Figure 8.5 have been combined with the weights appropriate to the amount of data in each, 3.7 and 7.4 respectively. The peak to central depression ratio for the two panels are 17.2 and 24.0, left to right. Contours are 5, 10, 20, 30%... of the peak in the global panel.

8.4.2 Restored Images

Finally, we present the restored super-resolved images with residuals added back in. Direct examination of the models is very useful, but can be misleading without an indication of the thermal noise. The relative scale between the residuals and model is something of a problem in super-resolution work, however. In normal imaging the “flux scale” of the residuals, such as it can be defined, agrees with the fitted beam. A point source entering through the residuals has the same height on the final map as a point source which has been converted to model components and back. Similarly, the integral over the near vicinity of a point source in the residuals is not terribly different from the integral over the same region of a point source in the components; both point sources have approximately the same width. Loosely speaking, the pixel values removed from the dirty map are divided by the volume of the dirty beam² when they are converted to model components, and multiplied by the volume of the restoring beam when they are transferred to the final image.

When the volumes of the dirty and restoring beam are appreciably different, there will be a flux scale mismatch between model components and residuals. The usual practice of simply adding the residuals back to the final image ensures that the peak value of point sources are correct, regardless of the relative sizes of the two beams. But if the restoring beam is much smaller than the dirty beam, the point source entering through the residuals will be too wide. Its flux calculated by an integral will be wrong. Worse, the numerical value of an extended source depends on the integral over the beam area. A given pixel in a perfectly smooth source which has been properly deconvolved and restored will have units of Janskys per restoring beam. If the restoring beam is R times the volume of the dirty beam, the numerical value of the extended emission drops by a factor of R . Smooth emission in the residuals, of course, will undergo no such transformation, and will be too high by a factor of $1/R$. The nature of the mismatch varies with the scale size of the residuals: compact residuals are numerically correct but too wide, smooth residuals are still smooth, but numerically too high. Notice also that if the program source is partially extended, its amplitude per beam area will drop after restoration with a super-resolved beam, lowering the apparent dynamic range relative to the noise. One can rescale the residuals to force the dynamic range to that measured from the nominal scale image, but this is ad hoc at best. Rescaling the residuals by the beam ratio R is correct in the limit of a smooth source, but this is probably too severe an attenuation of the residuals for most sources and difficult to justify. A slightly better alternative is simply to CLEAN the residuals to convergence over the entire image. CLEAN has the property that it can always reduce the residuals arbitrarily close to zero, barring a few caveats about numerical roundoff error. It can also be used over fairly wide areas with feasible computation times. When all the noise power as

²In this discussion, it may help to visualize the case where the dirty beam is a Gaussian with a well-defined volume. Essentially the same flux scale mismatch occurs as in the more complicated case.

well as the source power has been converted to CLEAN components, we may restore with any beam we like and everything will be on a consistent scale. The disadvantage is that the CLEAN process distorts the histogram of the noise considerably. CLEANed noise often looks speckley to the eye, even when restored at the nominal resolution, and the RMS value of the noise will no longer be equal to that calculated with the standard expressions. We no longer know exactly the noise scale is, other than that it is consistent with the source. For these reasons, I prefer to avoid CLEANing the noise when possible. In super-resolution work where the residual scale issue cannot be ignored, the practice is probably better than the alternatives. Making the best of a bad situation, the images presented here have all had the residual noise CLEANed to convergence. The residuals from the MEM or NNLS deconvolution are passed as input to CLEAN, which is then run without a support box until the final residuals are much less than the thermal noise. The component model used to create the restored image is the sum of the components from the first algorithm plus the CLEAN components. The restored image is created in the usual way from the original dirty image and the composite model, with no special scaling done on the near-negligible residuals. Figure 8.9 presents the restored image from the three major algorithms.

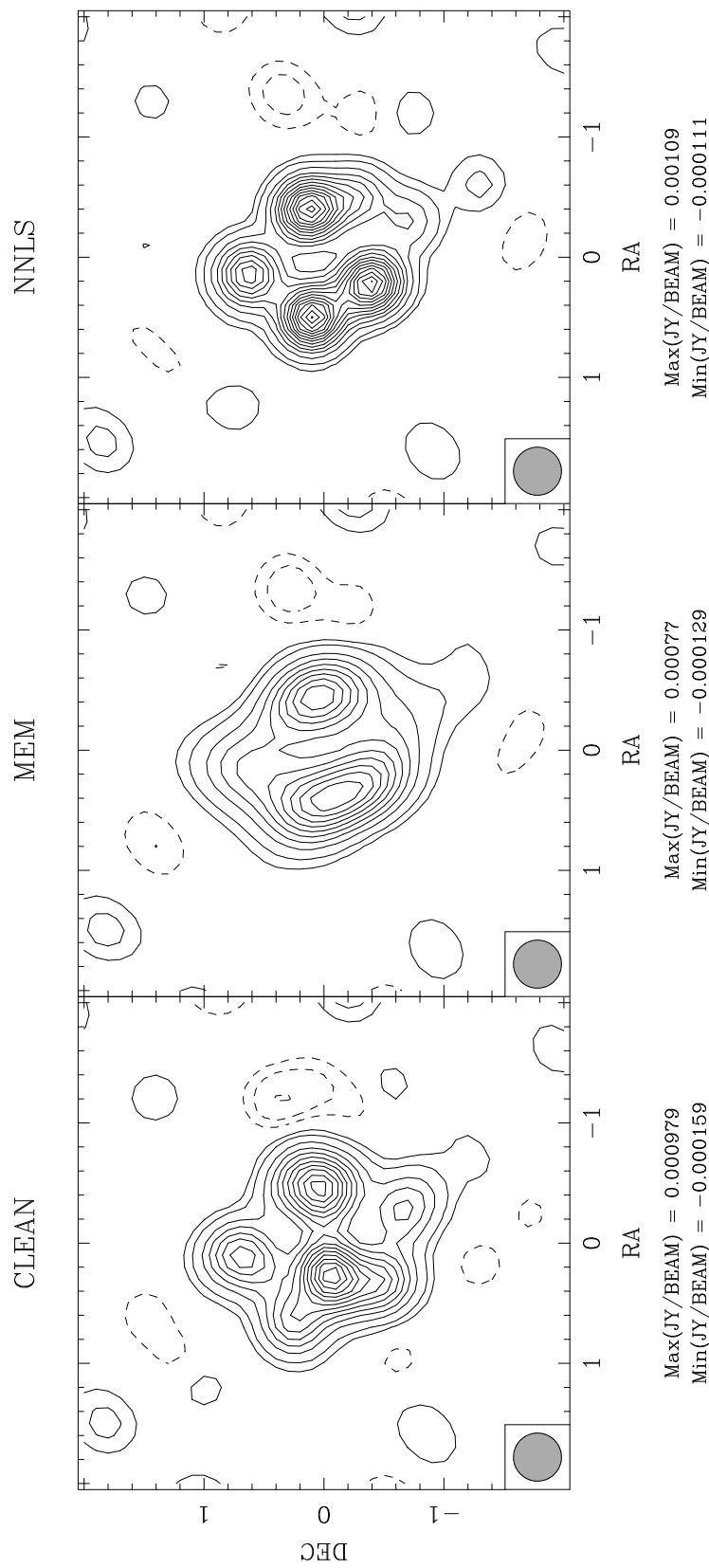


Figure 8.9: Super-resolved images against algorithm: the residuals have been CLEANed to $< 5 \mu\text{Jy}$ over the entire image before restoration, in an effort to maintain a consistent flux scale. The shaded circle is the $.4'' \times .4''$ restoring beam, not a fitted Gaussian to the PSF. Contours are common across panels at 5, 10, 20, 30% ... of the MEM peak.

8.4.3 Comparison with Other Data

Another good check on super-resolved features is comparison of the image with those from other telescopes or other wavelengths. Figure 8.10 shows the MEM restored images on a larger scale, superimposed on a greyscale image from the Hubble Space Telescope. The [O III] line used for the HST image is a primary atomic cooling line, and traces regions of dense gas. The image is from Jakobsen et al. (1991). The relative positioning of the two images is courtesy of a private communication from the Hipparcos astrometry group. There is no arbitrary shift involved: the relative positioning is completely *a priori*. The separation of the hotspots becomes more plausible, as it is symmetric around the supernova, and the nestling of the hotspots inside the circumstellar ring lends considerable support to the position angle being real as well. The hint of emission in a north-south direction along the polar axis is suggestive, but when one examines the residual image, both north-south extensions appear as connected components in a very noise-like and ordinary appearing residual image. Independent of what the supernova is doing, such an extension could have been produced by a slightly larger-than-average thermal fluctuation. This may be the start of a statistically significant feature in later epochs, but for the moment it should only be regarded as a suggestion. Notice on the radio images that the nominal resolution image has a 5% contour, and the super-resolved image does not. Clearly we have sacrificed considerable dynamic range in the name of resolution.

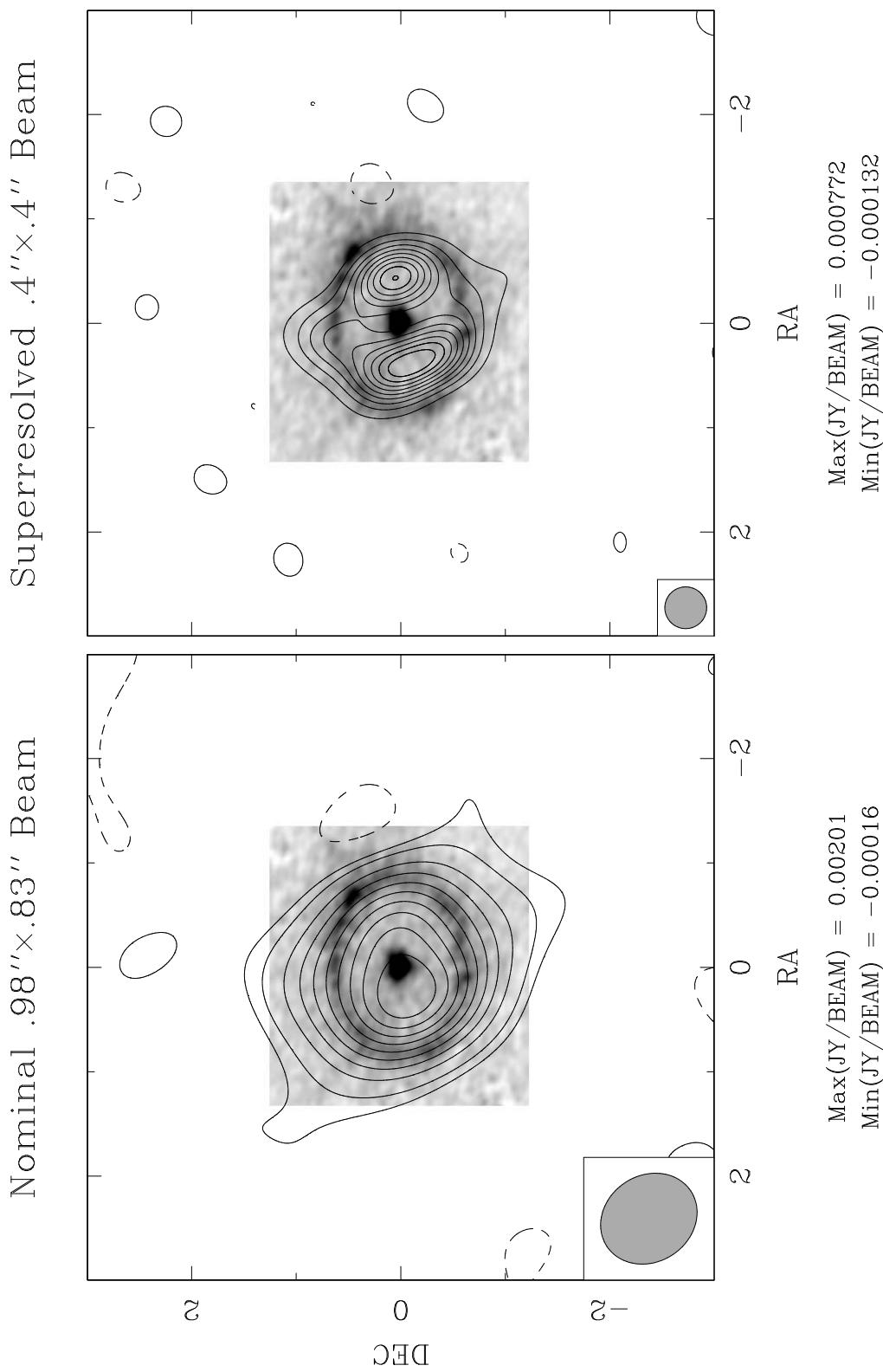


Figure 8.10: Comparison with other data: MEM restored images overlaid on the [O III] image from Jakobsen et al. (1991). The nominal beam radio image has contours 5, 10, 20, 30%,... of its peak. The super-resolved image has contours 10, 20, 30%,... of its peak.

8.4.4 Summary

There are a number of consistency checks on super-resolved images that can be performed on the observational data itself:

- Compare the results from different deconvolution algorithms, especially those with significantly differing properties.
- Compare results using different deconvolution parameters: In particular, check the image as a function of cell size over reasonable ranges of oversampling. Check that the result does not depend too sharply on iteration, though many algorithms have a long slow convergent tail. Check that the result does not depend too strongly on the details of the support constraints.
- Cross validate different subsets of the data. In this case, different epochs were imaged separately and compared. Also, random partitions of the composite data set were compared, which smoothed the calibration differences between epochs.
- Sum the deconvolutions from different subsets of the data. If the sum is similar to the global deconvolution of the combined data, algorithm nonlinearity isn't the driving force of the deconvolution and the solution is presumably more robust than otherwise.
- Compare the results carefully with other images at other wavelengths. Here we present our image superimposed on the 1991 [O III] image of Jakobsen et. al.

8.5 Visibility Weighting and Thermal Noise

A major portion of this dissertation discusses ways to optimally weight the visibility data, so it is only natural to try and apply the techniques of Chapter 3 here. Somewhat disappointingly, no weighting was found to be a significant improvement on the standard gridded uniform weighting for the purposes of super resolution.

In exploring the weighting parameter space, two strategies were employed. The goals were similar to the two usual regimes in robust weighting, but all available weighting parameters were utilized. The first strategy was simply to find a weighting that did everything possible to maximize the signal-to-noise ratio, with the secondary consideration of still gaining as much resolution as possible over natural weighting. The weighting field of view was increased somewhat to make up for the extreme oversampling. Gridless weighting was selected to make the best use of the signal, and a fairly high value of robustness was used. A good choice was found to be $FOV_w = 2$, a robustness of 0.7, and gridless weighting. The RMS thermal degradation was only 1.042, a dramatic improvement over the uniform 1.730, and the beamsize of $1.187'' \times .886''$ was also an improvement over the natural $1.324'' \times .967''$. Clearly some resolution has been lost over the uniform weighting $.968'' \times .832''$, but it seems a reasonable tradeoff for the extra sensitivity. As a pleasant side effect, the worst negative in the beam has risen from the uniform -14.4% to -12.3% , though this is really coming from a natural weighting plateau that is starting to form.

The second strategy was to seek the tightest possible fitted beamwidth, while degrading the noise as little as possible. A very high degree of super-uniform weighting was selected, plus an inverse taper. Gridless weighting and a moderate amount of robustness was used to pull the noise back to a reasonable value. Eventually the parameters $FOV_w = 1/5$, Taper = $-.5''$, Robust = 0, and gridless weighting were selected. From the beam parameters, the compromise seems to have worked quite well. The beamsize has dropped to $.870'' \times .762''$, while the thermal degradation was even lower than the uniform value at 1.349. The price which has been paid is that the most negative sidelobe has become worse at -21.6% . Slices through the major and minor axes of all of these point spread functions, the low-noise PSF, the normal uniform PSF, and the high resolution PSF, are given in Figure 8.11.

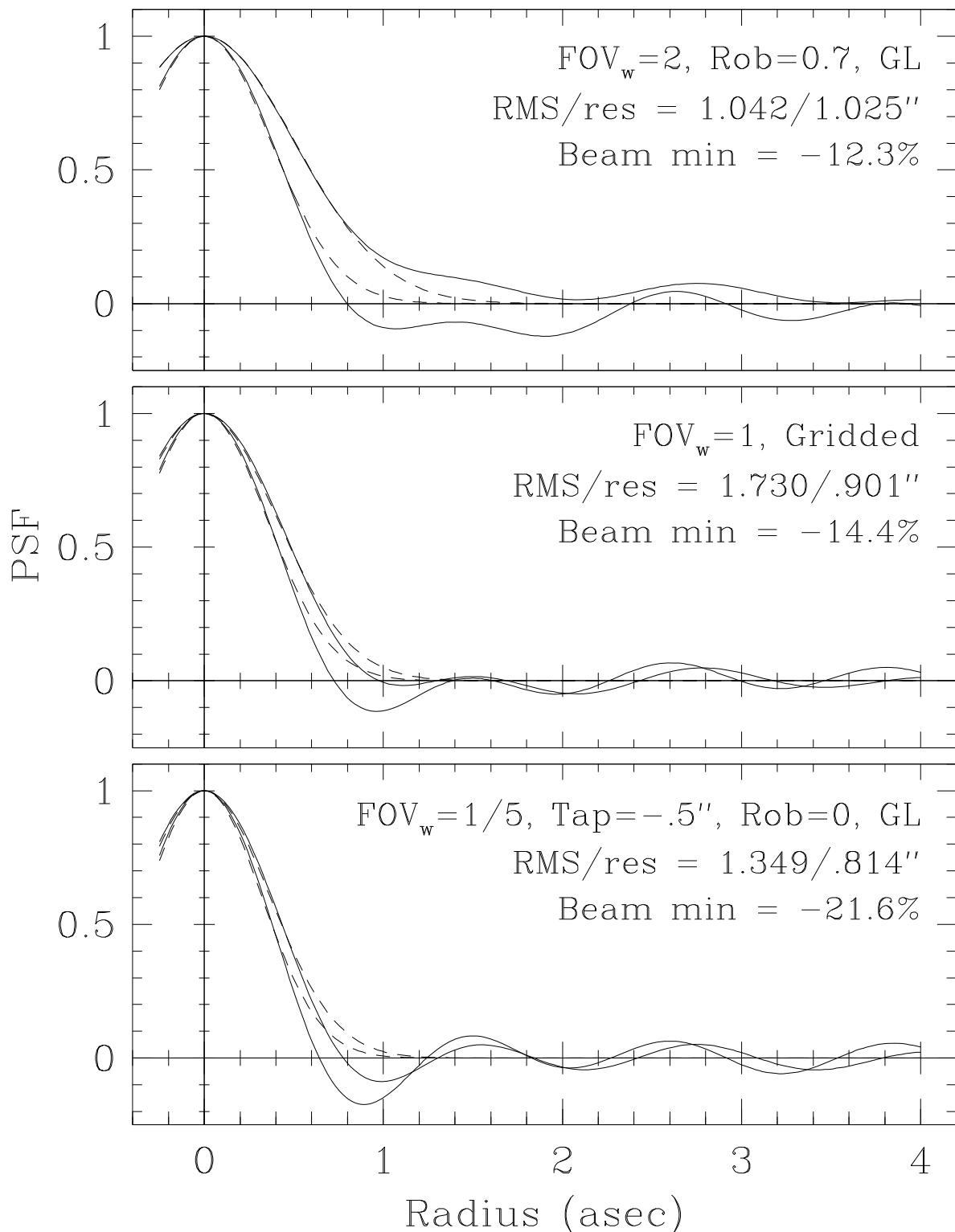


Figure 8.11: PSF slices against weighting.

The MEM sky models for the different weightings are given in Figure 8.12a, and the NNLS models in Figure 8.12b. As described in the next section, the average effective resolution of the MEM model was estimated to be roughly $.4''$, so the NNLS models were smoothed down to this resolution for comparison and ease of interpretation.

The weighting results are disappointing, and there seem to be two causes for the failure of the exotic weightings to improve the reconstruction. The first is simply that this is a fairly crowded field in terms of low level emission, both extended and compact. (The supernova is in the Large Magellanic Cloud, so HII regions are fairly common.) Large maps of the entire primary beam show a number of faint compact sources, many in the hundreds of μJy , scattered evenly throughout the region. Presumably there is a distribution of such sources, with many more comparable to the noise, and not so easily identified. There is large scale structure evident in the maps as well, though it is not as clear whether this is astrophysical or instrumental in origin. The one thing that is abundantly clear is that this will not be an easy field to image down to the thermal limit. A fairly straightforward demonstration of this is that the measured off-source RMS on the standard 256^2 nominal resolution NNLS image centered on the supernova is $51.1 \mu\text{Jy}/\text{beam}$ for uniform weighting, compared to the predicted $43.9 \mu\text{Jy}/\text{beam}$. Going to natural weighting, the noise actually *rises* to $53.3 \mu\text{Jy}/\text{beam}$, instead of dropping towards the predicted $25.4 \mu\text{Jy}/\text{beam}$. This statement is true of both the MEM and NNLS images.

The confusion was actually diagnosed prior to mapping the entire primary beam, and even before the high noise in the naturally weighted image was noticed, by the technique of comparing median and mean filtered data discussed in Section 6.1. The confusion also shows up as significant variations in the estimates of ΔS_{uw} from different subsets of the data and from boxcar averages with different averaging times. (The noise statements above are based on a successive difference determination from the entire data set. Arguably, it should be adjusted downwards by 5 to 10% to account for the confusion, resulting in a still larger discrepancy between measured and expected thermal noise.) And the confusion also manifests itself as higher than expected χ^2 statistics between models and the u - v data, with very nonstatistical variations as different data are allowed into the fit. It all hangs together quite nicely. When analysed carefully, noise statistics can be quite indicative of the physics behind the limiting factors to the imaging.

No attempt was made to CLEAN the entire primary beam at $.1''$ cellsize, though if this data set is re-processed at a later time when computers are significantly more powerful, something might be done about reducing the confusion. For the moment, we must simply live with a higher than thermal noise, and note that it does not seem to be affecting the project significantly. From the point of view of super resolution, this confusion problem means that improving the thermal noise properties of the PSF does not directly result in a signal-to-noise improvement. The far sidelobes of the gridded uniform beam are difficult to improve upon, even with robust weighting. Consequently the first of the two robust

weightings has increased the size of the main lobe for no particular benefit. In this light, it is not surprising that the effective resolution of the MEM model should decrease.

The second weighting attempted to improve the resolution of the deconvolution model by sharpening the main lobe of the primary beam. As mentioned before, this was fairly successful from the point of view of the PSF statistics. The PSF is significantly sharper than the uniform case, and the thermal noise degradation is also significantly better. This weighting actually does produce the lowest measured image RMS of any weighting considered, at $42.9 \mu\text{Jy}/\text{beam}$ for the nominal images. This is probably due to the emphasis on the higher spatial frequencies filtering out the confusion noise somewhat, as well as to the improved thermal degradation factor. (The confusion noise is most significant at the lower spatial frequencies, though present at all frequencies. One can throw away the inner 35% of the data by $u-v$ radius and *decrease* the measured RMS of the naturally weighted image from $53.3 \mu\text{Jy}/\text{beam}$ to $51.8 \mu\text{Jy}/\text{beam}$.) Unfortunately, the improved main lobe and improved thermal RMS does not translate into higher resolution. The price that was paid for this weighting was the increase in the first sidelobes from -14.4% to -21.6% . This alone seems to prevent MEM from achieving the highest effective resolution. More work could be done with modelling to explore this behavior.

Notice in Figure 8.12 that the NNLS models seem more robust to variations in the weighting than the MEM models, as determined by the variation or lack of it in the central depression. Varying the weighting does change the relative importance of the statistically-independent visibilities, and might be expected to alter features in the image that are determined only by statistical fluctuation. The questionable extension to the north, for instance, is varying in strength somewhat, as is the elongation or breakup of the eastern hotspot. One does not know whether to claim that the comparative stability in these features between NNLS and MEM is evidence that they are real, or that the still quite significant variation in these same features in the NNLS panels is evidence that they are not. More data is certainly needed to answer this question.

As a whole, the variation of the deconvolution models with weighting was not terribly significant in this project. A cautious investigator might wish to explore different weightings in a similar project, but it seems that in cases where the change in thermal sensitivity between weightings is dominated by confusion or super-resolved deconvolution errors, the conventional gridded uniform beam does at least as well as other choices. It remains to be seen how well this result will apply to cases where confusion is not an issue.

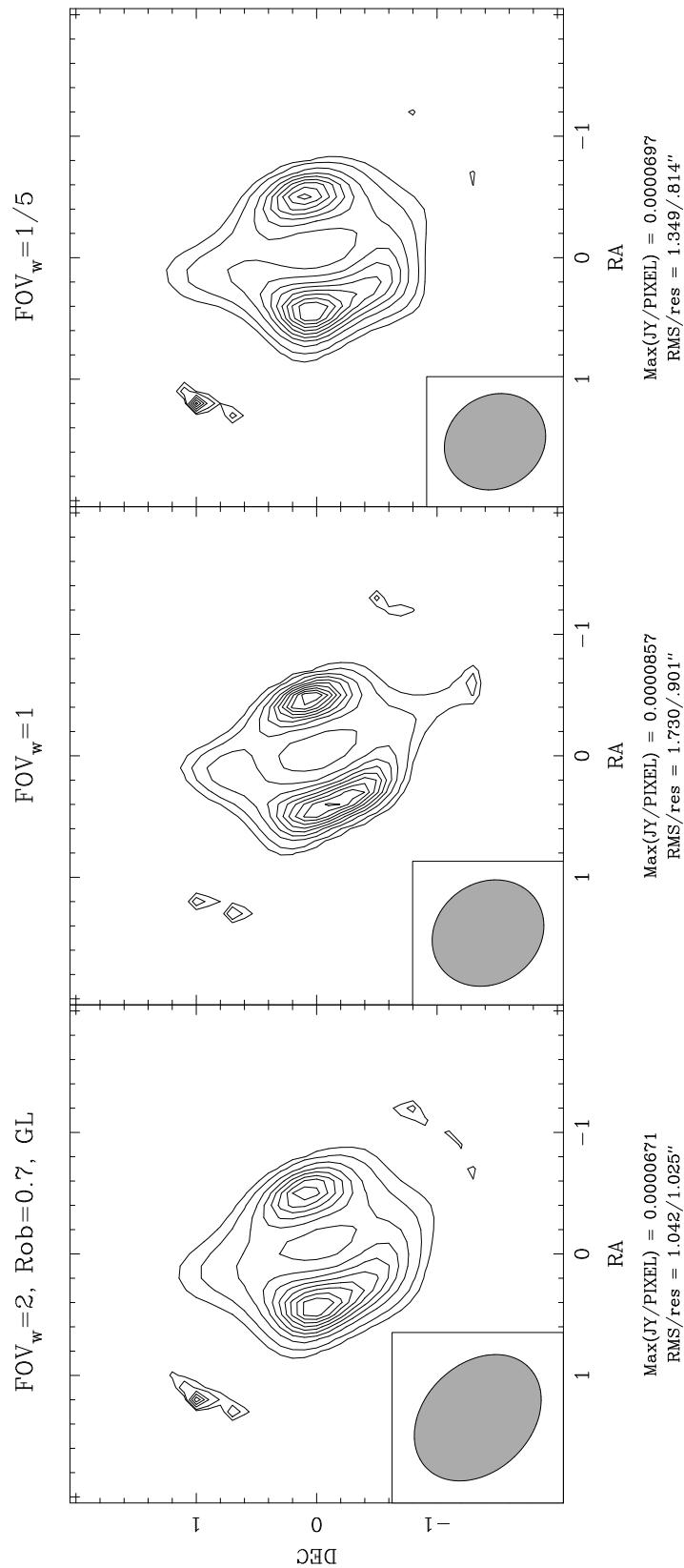


Figure 8.12a: MEM sky models against weighting. The ratio of peak to minimum central value is 6.5, 17.2, and 7.0, left to right. The shaded ellipses are the fitted Gaussian to the PSF. Contours are 5, 10, 20, 30%,... of the peak in each panel.

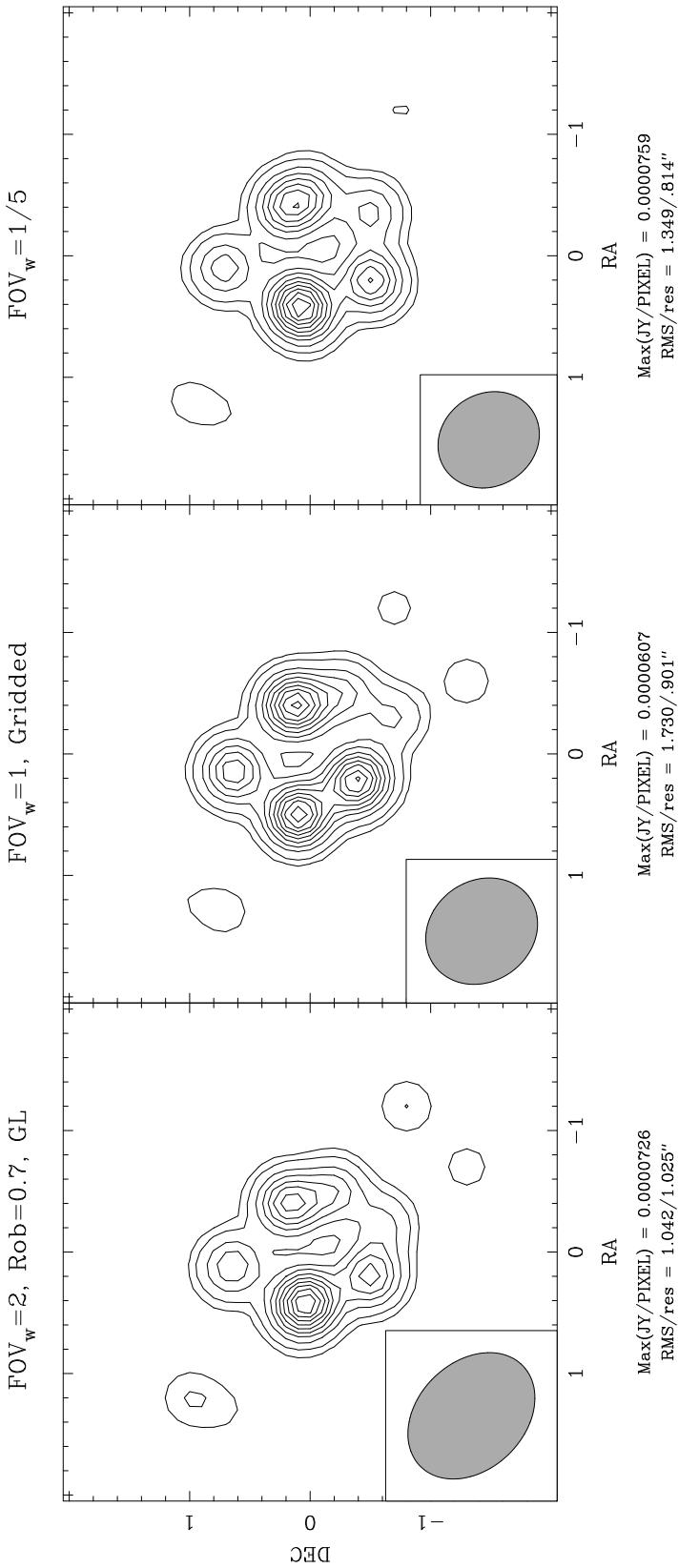


Figure 8.12b: NNLS smoothed sky models against weighting. The model components have been smoothed by a circular Gaussian of width $.4''$. The ratio of peak to minimum central value is 13.0, 12.0, and 16.5, left to right. The shaded ellipses are the fitted Gaussian to the PSF. Contours are 5, 10, 20, 30%,... of the peak in each panel.

8.6 Modelling

Comparison of deconvolution algorithms and variation of imaging parameters is a good first step in assessing the reliability of a super-resolved observation, but one always wonders if some pathology in the data itself is driving all of the deconvolutions to a common wrong answer. The position angle of the PSF is an obvious example — image features are commonly extended at that position angle, even though the deconvolution model is free to first order of sampling artifacts. This particular data set suffers from a problem commonly found with east-west interferometers. The rising and setting of the source has left a substantial wedge shaped region of the u - v plane unsampled, as shown in Figure 3.12b. This is the reason that robust weighting cannot find a good Gaussian-like PSF; the necessary robustness is very different along the axis of the missing data than along the orthogonal axis. It is unfortunate in this case that the elongation of the PSF lies along the direction of a possible extension to the north-east, and is almost exactly orthogonal to the position angle of the hotspots. There is definite cause for concern that this is affecting the reconstruction.

The only feasible way to assess the reliability of super-resolved images like these is by modelling. (Modelling here refers to the model used to generate a simulated data set, not the implicit modelling performed by the deconvolution algorithms.) Direct fitting of the models to the u - v data, bypassing the deconvolution step all together, is a good idea when possible. In this case, the model-fitting itself is a form of deconvolution. Here, we have used u - v plane model-fitting to solve for such parameters as the radius of a spherically symmetric model at earlier epochs than shown here. The single or few parameter models seem quite robust, and analyses using this procedure on this and earlier data sets are given in Staveley-Smith et al. (1993a and 1993b). Unfortunately, the confusion problem mentioned earlier becomes more serious as the model becomes more complicated, especially for u - v plane model-fitting. The smaller variations in the model χ^2 parameter used to differentiate between the differences in detailed models becomes washed out by the errors due to confusion noise. It was found that models with a spherical shell and two hotspots were only mildly favored by the u - v model-fitting over a single hotspot, though the second hotspot is clearly mandated by the imaging. Detailed placement of two or three hotspot point sources was not possible with u - v plane modelling. Restricting the data entering the fit to the longer spacings helped somewhat, but not enough to be useful.

The alternative, then, is to model in the image plane. The essential difference is that image plane modelling partially decouples the confusing sources from the model. The confusing sources enter the modeling only via their sidelobes, and not directly through the Fourier sum. Obviously, the sum of the squared model residuals is the same in the image or the u - v plane, but the distinction is that in the image plane one can choose to consider only the residuals around the source of interest. By fitting in the u - v plane, one is fitting simultaneously to all the emission in the primary beam.

While one could conceivably automate the process of deconvolution and feature extraction, and then actually fit for the parameter values in the conventional sense, it would

be extremely computer intensive. Each ‘function evaluation’ of the optimization criterion would involve a full deconvolution. This might one day prove useful, but was not attempted in the current project. Instead, a small number of Monte Carlo style simulations were run to gain an understanding of how sensitive the super-resolved imaging process is to statistical fluctuations and sampling bias.

The forms examined were a spherically symmetric Gaussian, an optically-thin sphere of emission, an optically thick sphere (a disk in projection), and an optically-thin shell of infinitesimal thickness. The shell was the closest match to the data, and shells with one, two, three and four delta function hotspots were examined. Finally, the best MEM model found was iterated back into the process as a model source. Visibility data were generated from each model and noise added which matched that measured in the image plane. The noise was purely Gaussian in nature — no attempt was made to model the confusing sources, and no calibration errors were simulated. The general conclusion from all of these tests is that the most basic features of the image are quite robust.

By integrating the regions of the hotspots in the best MEM model, and crudely accounting for the presumed thin shell beneath them, we came to the conclusion that the hotspots contained roughly 50% of the total source flux. As evidence that the hotspots are indeed real, consider Figure 8.13. The model here is a thin shell with 4.49 mJy total flux, roughly the same flux as the supernova, and diameter 1.30''. Smoothing effects cause the radius of peak flux on the image to be slightly less than the shell radius. The deconvolution and thermal noise is indeed creating spurious peaks in the model image, and they do resemble the hotspots seen in the real image. However, the average total integrated flux in the spurious hotspots seen was of order 5% — a factor of ten less than seen in the real image. Five realizations of the thermal noise were examined. The model images converge more quickly than the real data, and only 200 MEM iterations were needed to reach convergence.

The importance of the PSF in determining the measured position angle of the hotspots was tested by creating new synthetic data sets from the original, rotating each u - v point by a fixed angle and calculating the model visibility at the new point. That is, we held the model fixed on the sky, and rotated the point spread function. Again, thermal noise equal to that seen in the observation was added, and four realizations of thermal noise examined. The shell plus two point model was used, but an extremely conservative flux for the point sources was used, 16% of the total flux instead of 50%. The flux in the shell, eastern hotspot and western hotspot was 4.39, .64, and .22 mJy, respectively. The PSF was rotated by an angle of 0°, 45° and 90°. The resulting MEM models are given in Figure 8.14. Clearly there is significant variation in the measured position angle between the image peaks, but it is much less than the angle of rotation. Remember that this test used a very conservative flux in the hotspots. As more fractional flux is placed into the delta function components, presumably the position angle extraction will become more robust still.

The simulated observations can also be used to determine the effective resolution of the MEM model. The algorithm has a bias towards smoothness, and will only make a

peak as sharp as needed to agree with the data within the noise. Also, higher peaks will in general be made sharper than lower peaks. Consequently MEM has a (space variant) internal resolution in the sky model. In deconvolutions where the true answer is known, an effective mean resolution can be determined by simply smoothed the original model for the visibility data until it best matches the MEM component model. This procedure was performed on the shell + 2 spot model described above and repeated for 4 realizations of the noise. The resolutions of best fit were determined only roughly, but in all cases were between $.3''$ and $.4''$. The justifies the statement that the final image smoothed by a restoring beam of $.4''$ will have an effective resolution of approximately $\sqrt{.4^2 + .35^2}'' \approx .5'' - .55''$.

Summarizing the basic strategies for using model-fitting to verify super resolution of simple sources, we suggest

- Select plausible models from initial images.
- Modelfit for parameters, using the original $u-v$ data directly if possible.
- Check that simulated observations can recover the significant features of the model reliably, when present in the synthetic data.
- Run plausible models less the features in question though the imaging cycle — can the features be produced spuriously? At what level?
- Vary the parameters of the observation to alter the PSF significantly. Rotation of the $u-v$ sampling is particularly useful to check for position angle bias.
- Use simulated observations to determine the effective resolution of the deconvolution sky model.

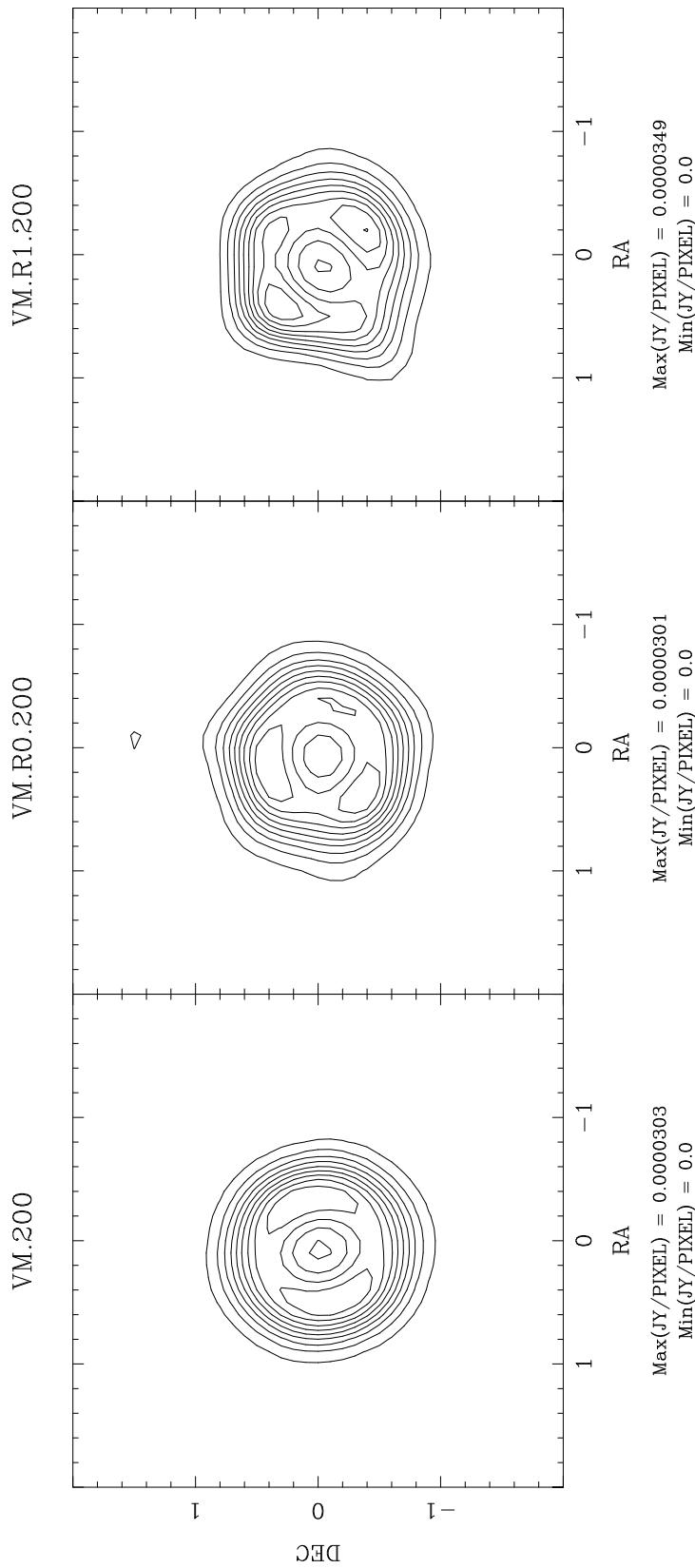


Figure 8.13: Modelling of spurious features: maximum entropy models demonstrating that spurious peaks can be produced by the deconvolution process. The simulated data is derived from an optically thin sphere of emission. All peaks along on the ridge are spurious. The leftmost image is noise free, and the other two images differ only in the particular realization of the simulated noise added. The measured flux in such spurious peaks was less than 5% of the total — much less than the $\sim 50\%$ found in the actual observations.

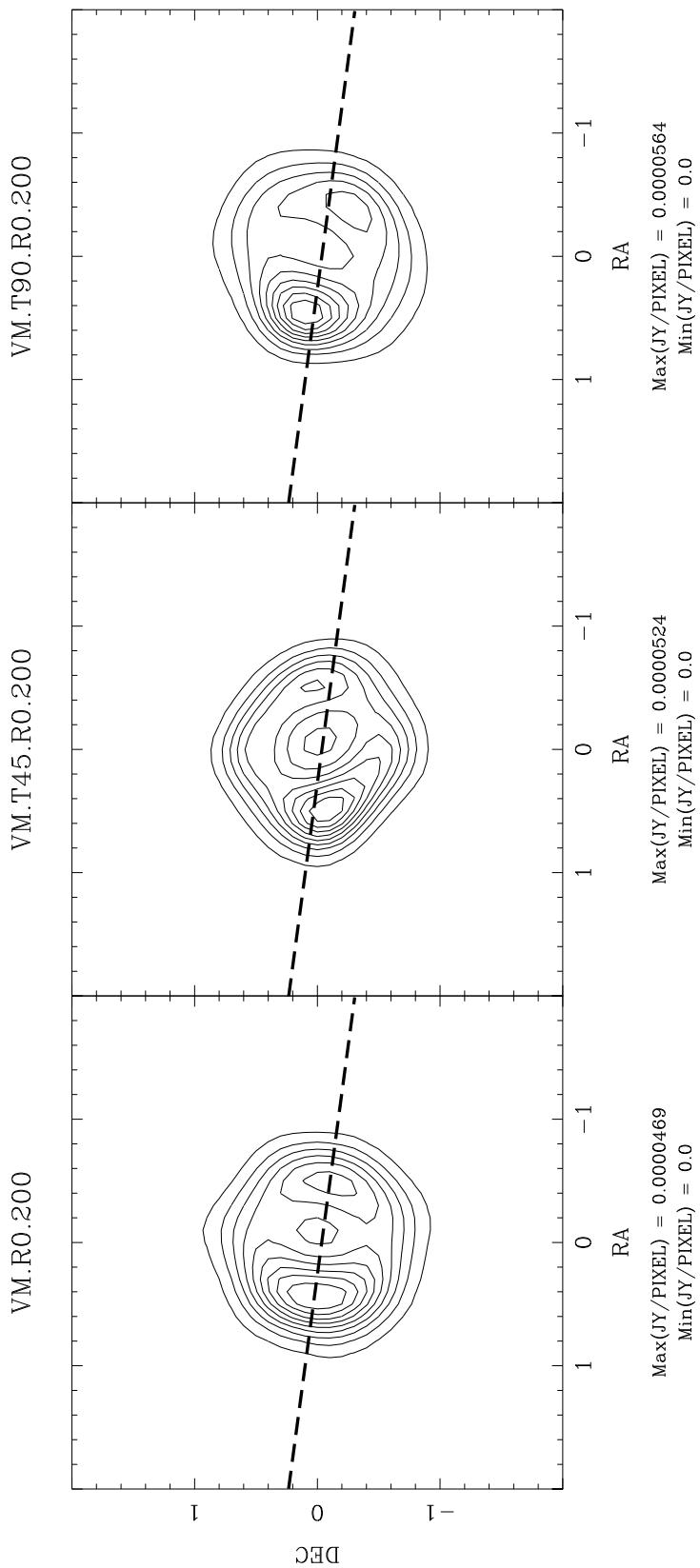


Figure 8.14: modelling of PSF position angle bias: the PSF is rotated while holding the model position angle fixed. The model is an optically thin shell plus two point sources and has only 16% of the flux in the point sources — probably a conservative lower limit. Thermal noise has been added, so the real reconstruction is probably at least as reliable as this. The dashed line connects the two model point sources. The PSF is that of the original observation and has been rotated by 0° , 45° and 90° . While not constrained closer than perhaps $\pm 10^\circ$, the position angle of the spots is clearly not driven by the PSF alone.

8.7 Results and Discussion

We have come to the following conclusions about the features in the super-resolved image:

- The central depression is real. It is difficult to be quantitative about the peak-to-depression ratio at the effective $.4''$ – $.5''$ resolution, but factors of at least 5 to 10 appear mandated by the data.
- The hotspots are real, and contain a significant portion — perhaps 50% — of the total flux.
- The position angle of the hotspots agrees well with the FOC image, which is encouraging, but it is not particularly well constrained by the radio image alone.
- There is a suggestion that the eastern hotspot is elongated in the north-east direction, or that two compact components might be present there.
- The suggestion of emission along the polar axis is not yet judged to be significant, as there is suspicious correlation between the MEM sky model and the residual image.

Correlation of these last two features with future observations may help resolve the current ambiguity. General recommendations and comments about the algorithms used are:

- MEM is the best algorithm found for super resolution in this study. NNLS can also produce good super-resolved images, but requires that the component model be smoothed prior to interpretation. The compensating factor is that NNLS seems less susceptible to variations in weighting. In difficult cases, one is wise to examine both algorithms.
- GSP, Richardson-Lucy, and Maximum Emptiness are well behaved but suffer in robustness or resolution. CLEAN and SVD algebraic inversion are not recommended for super resolution.
- When using MEM for super resolution, many more iterations are required than for normal suppression of sidelobes. Convention wisdom holds that 40 iterations is usually sufficient. In this study, the image was still sharpening at 2000 iterations.
- The effective resolution of the MEM model is a strong function of the close-in sidelobes of the PSF. A sharper main lobe in the PSF does not always lead to a higher resolution model, even apart from thermal noise sensitivity considerations.

As to the astrophysics of the source, the radio emission is generated by the shock between the supernova ejecta and the surrounding material; radio observations can consequently be used to deduce the mass loss history of the progenitor star. Here, we believe that we are seeing an edge brightened torus of emission, aligned with the equatorial plane of the

remnant defined by the circumstellar ring seen in [O III]. The progenitor star of SN1987A spent considerable time as a red supergiant, with a dense, slow stellar wind. Unlike most type II supernova, the progenitor was actually a blue supergiant with a much faster, less dense wind. This last phase of the star's history before the explosion is thought to be short, having lasted perhaps $2(10)^4$ years. The fast wind from the blue supergiant swept up the slower red supergiant wind, creating the density enhancement that became the circumstellar ring. The region interior to the ring is a reversed shocked interaction region between the red and blue winds, and still interior to that is a region where the blue wind is in free expansion. The expanding shock wave from the supernova is currently in the red-blue wind interaction region, previously thought to be of roughly-constant density. That the radio emission is coming from a torus would indicate that the density in the equatorial plane is higher than in the polar regions and that the overall density of this interaction region is not as constant as previously believed. If the shock front is significantly decelerated by the enhanced density in the equatorial plane and continues at the present velocity in the polar directions, we may possibly be witnessing the start of a classical "barrel-like" supernova remnant. Our full astrophysical interpretation of these images can be found in Staveley-Smith et al. (1993a and 1993c).

Chapter 9

Case Study: DA193

This chapter is a small collection of related material about the source DA193. Section 9.1 is a near verbatim transcription of VLBA memo #697A, (Briggs et al., 1994), written in collaboration with R.J. Davis, J.E. Conway, and R.C. Walker. Figure 9.1 showing the details of the observation parameters has been added, and the memo has been formatted according to the conventions of the dissertation. While some of the material in the memo is not surprising in the light of the rest of this dissertation, it does provide an interesting look at an early practical use of NNLS deconvolution, and the first observational example of the comparatively sparse VLBA u - v coverage. It also provided the motivation for Section 9.2, consisting of simulations of the entire hybrid mapping loop, both deconvolution and self-calibration. The simulation section is important in a practical sense not only for the conclusion about the robustness of NNLS, but also for demonstrating the importance of gain correlation structure in realistic simulations. A simple fix to existing practice is given for post hoc modelling of existing observations. Finally, in the concluding comments of Section 9.3 we provide a glimpse of wonderful things to come with an image from work in progress by Tim Cornwell – extremely high dynamic range imaging of DA193 with nearly the full VLBA bandwidth.

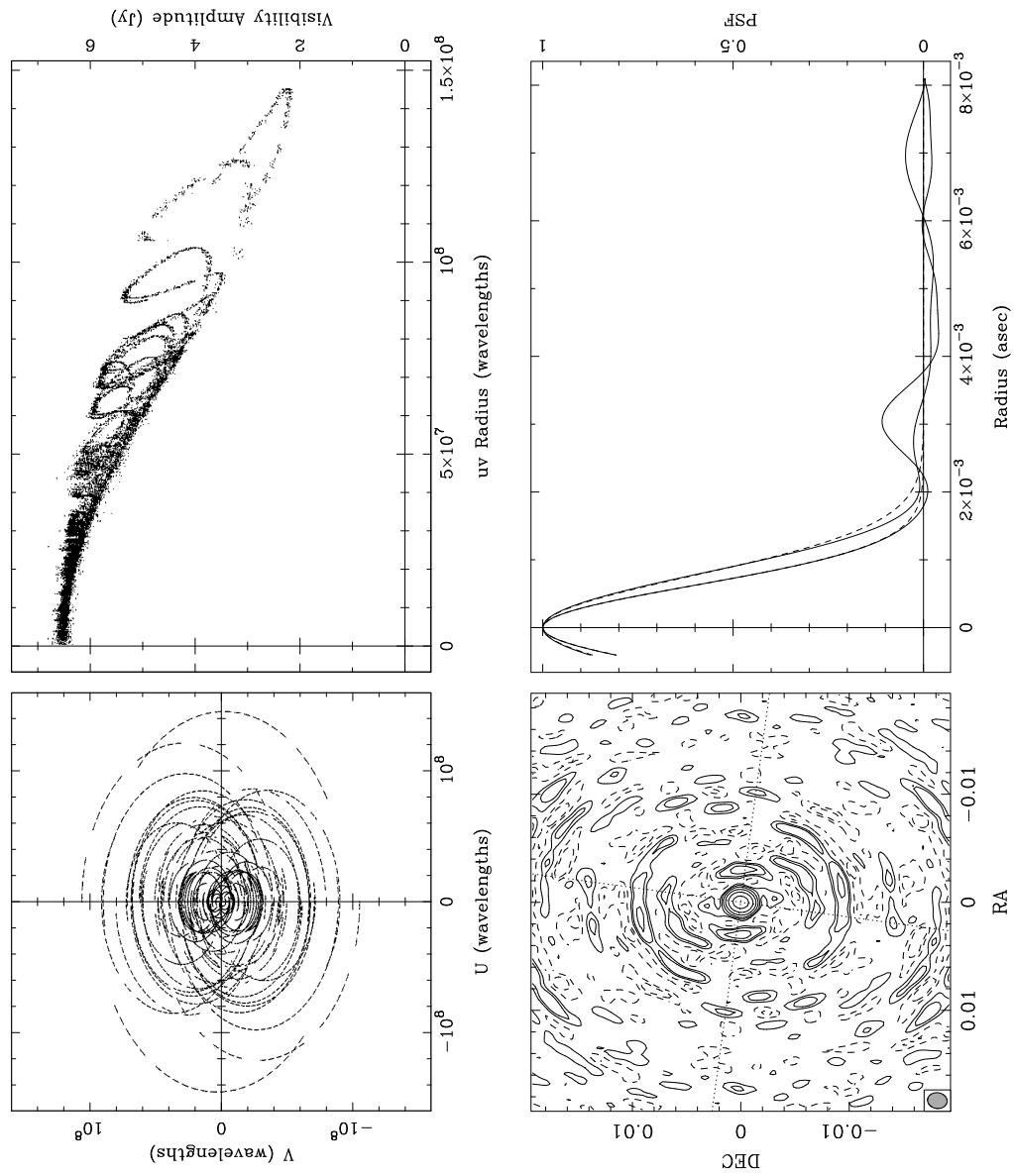


Figure 9.1: Observational and imaging parameters.

9.1 VLBA Array Memo #697A

High Dynamic Range Imaging With the VLBA

D.S. Briggs

R.J. Davis

J.E. Conway

R.C. Walker

National Radio Astronomy Observatory

July, 23 1994

9.1.1 Introduction

The VLBA is being tested for high dynamic range imaging using a complete track on DA193. All the data have been fringed, calibrated and edited by Craig Walker. Here we report on attempts made to image just one of these IF channels. In principle this data set had the possibility at being imaged at $\gtrsim 28,000:1$ dynamic range. This is well beyond current limits achieved in VLBI. Furthermore the VLBA has the potential in terms of signal to noise to exceed dynamic ranges of 150,000:1. At this level we are concerned about nonclosing errors in the correlator and station hardware, self-calibration errors, and deconvolution errors. Our current work achieves a measured dynamic range of 27,700 and extremely good agreement with the a priori thermal noise. The best images show no sign of any hardware effect or processing instability. Significant artifacts are seen in the CLEAN only deconvolution image, but several other techniques are investigated with successful results.

9.1.2 Initial Calibration

The experiment consisted of full tracks on DA193 at C band, for a total observation of 17 hours. Data were recorded for 13 minutes out of every 18. The observation was single polarization, RR, and after bandpass calibration the effective bandwidth was 1.75 MHz.

The initial calibration was done in AIPS. The amplitude calibration was done using the a priori gains and system temperatures. Since the overall scaling factor was not known, it was adjusted to force the total flux density of DA193 to match that measured in a special pointing run made a few days before. Large gain adjustments were required for Pie Town (pointing problem) and Hancock (snow). The data were edited based on a priori edit information provided by the VLBA system and based on system temperatures and known antenna problems. Once all of these edits were done, and low weight points

were removed, no further editing was required. Fringe fitting was done with **FRING** and a bandpass calibration was done after fringe fitting. The bandpass calibration was done after fringe fitting so that proper amplitude corrections would be made, to the fully calibrated data, for amplitude losses due to the phase slopes caused by delay offsets. Finally the data were averaged in frequency, omitting the end channels, and in time. No baseline dependent (closure) calibrations were made.

9.1.3 CLEAN Deconvolution

Initial attempts to image IF 3 were made with CLEAN, strongly favoring only positive components. This produced images with clear rings of emission around the source and covering the entire map. Allowing negative components solves this problem and produces the maps of Figure 9.2.

The extended halo with twin banana like structure is similar to the dirty beam and is probably nonphysical. At this stage we were worried about the possibility that this could have been caused by closure problems. To this end we processed IF 7 through the same software route. This IF channel has been through completely different hardware on its route to the workstation disk. The two IFs passed through different filters at the stations, different samplers, tracks on the tape, decoding at the correlator, delay lines, phase rotation, correlator chips and so forth. However, within mapping errors which are bound to be a few times noise given the selfcal correction, these images are identical. Subsequent images were all made from IF 7.

Examination of the CLEAN components shows 4 alternate rings of positive and negative CLEAN components. Studies and simulations, [this dissertation], have shown this to be due solely to the CLEAN deconvolution algorithm. Tight CLEAN windows actually make the solution somewhat worse. The peak of the extended emission in these images is roughly 0.1%. The dynamic range is 9000:1, worse than the theoretical of 14500:1. It was clear that a better means of deconvolution is required, especially when the VLBA is used at the full recording rate of 128 Mbits/s instead of the effective 3.5 Mbits/s used in this experiment. Several additional deconvolution/calibration strategies were explored, all of which were successful.

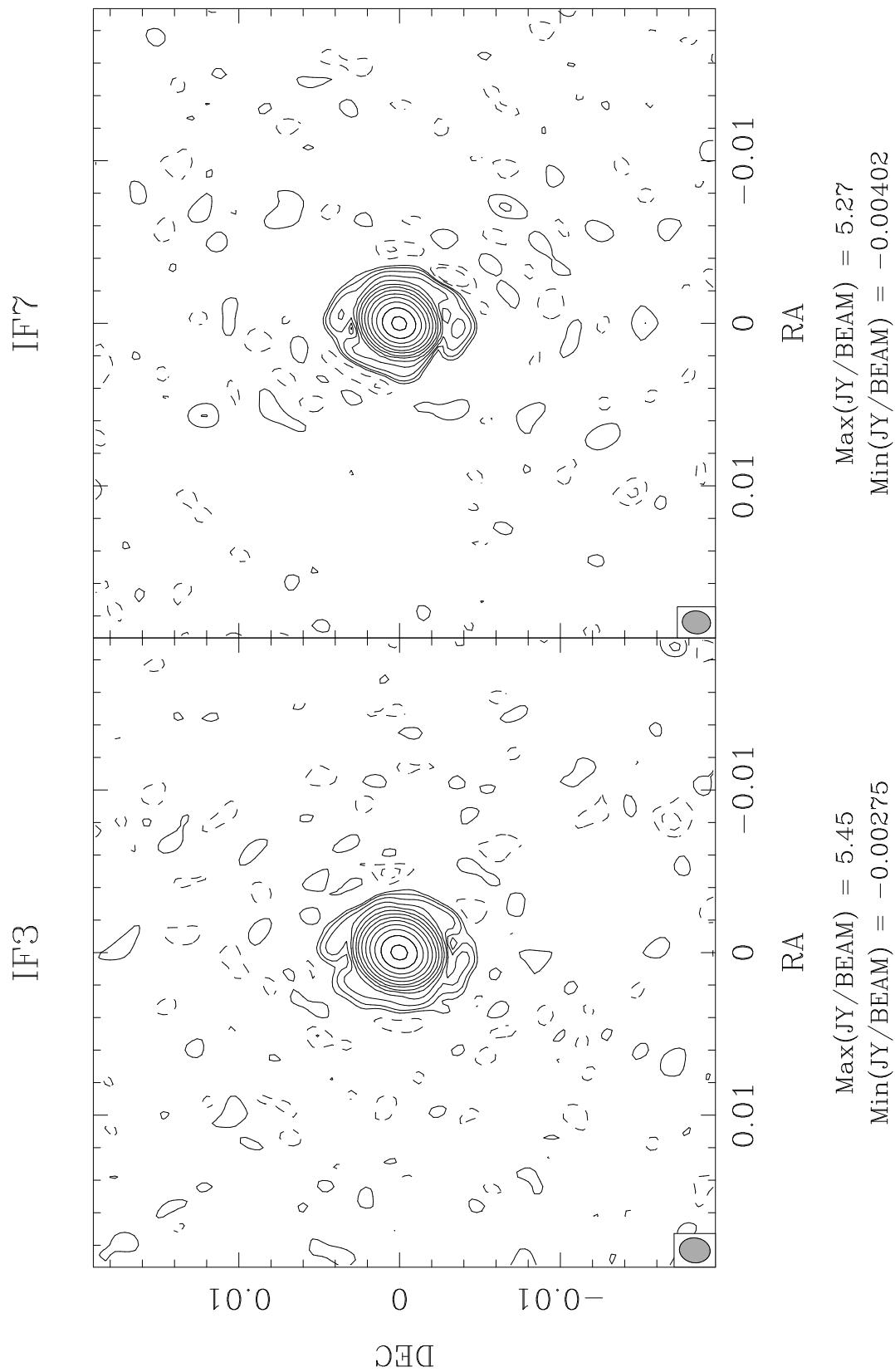


Figure 9.2: CLEAN images of two different IFs produced in `difmap` by RJD. Peak to off-source RMS is 9000 and 9400 for IFs 3 and 7 respectively.

9.1.4 Imaging Presentation and Metrics

In situations where there is a significant mismatch between the main lobe of the beam and its fitted Gaussian, the issue of the residual flux scale is important and directly effects the meaning of “dynamic range.” The images presented here have all had the residual noise cleaned to convergence. This practice does distort the noise histogram somewhat, and it also causes the noise RMS to deviate from that calculated with the standard expressions. (This is not surprising, since the units of these expressions are strictly Janskys per dirty beam, and the whole point of cleaning to convergence is to change units to Janskys per restoring beam.) However, so far as is possible the restored model and noise are on a common scale and this is our justification for quoting dynamic ranges based on the cleaned noise images. Dynamic range in our usage is the peak in the image over the off-source RMS in the cleaned noise images. By contrast, our noise comparison with the a priori sensitivity calculations are based on the off-source RMS in the images without noise cleaning. For the naturally weighted fitted beam of 2.7×2.3 milliarcseconds, the noise cleaned off-source RMS drops by 19% compared to the image where only the source is cleaned. For the uniformly weighted beam of 1.8×1.5 mas it drops by 28%. For the super resolved restoring beam of 1.0×1.0 mas, it drops by 47%. Thus while our definition of dynamic range is self consistent, measured values can exceed the nominal values derived by simply dividing peak source strength by thermal noise. Our theoretical dynamic ranges have been corrected for the noise rescaling effect.

All images presented here are contoured logarithmically, with a factor of 3 between contours. Unless stated otherwise in the caption, the lowest positive contour is .02% of the peak.

9.1.5 Model Fit with CLEAN Tidying

John Conway has fitted the bulk of the emission with a two component elliptical Gaussian model. The residuals from the modelling processes were then processed with CLEAN. This image was then used as the basis for one iteration of self-calibration and remapped with the same model. The result is Fig 3 which shows no artifacts in the noise and a smaller extended region around the source. It seems likely that further work with self-calibration might improve the image still, but this has not been done. This image is given in the left half of Figure 9.3. The dynamic range is 10500:1.

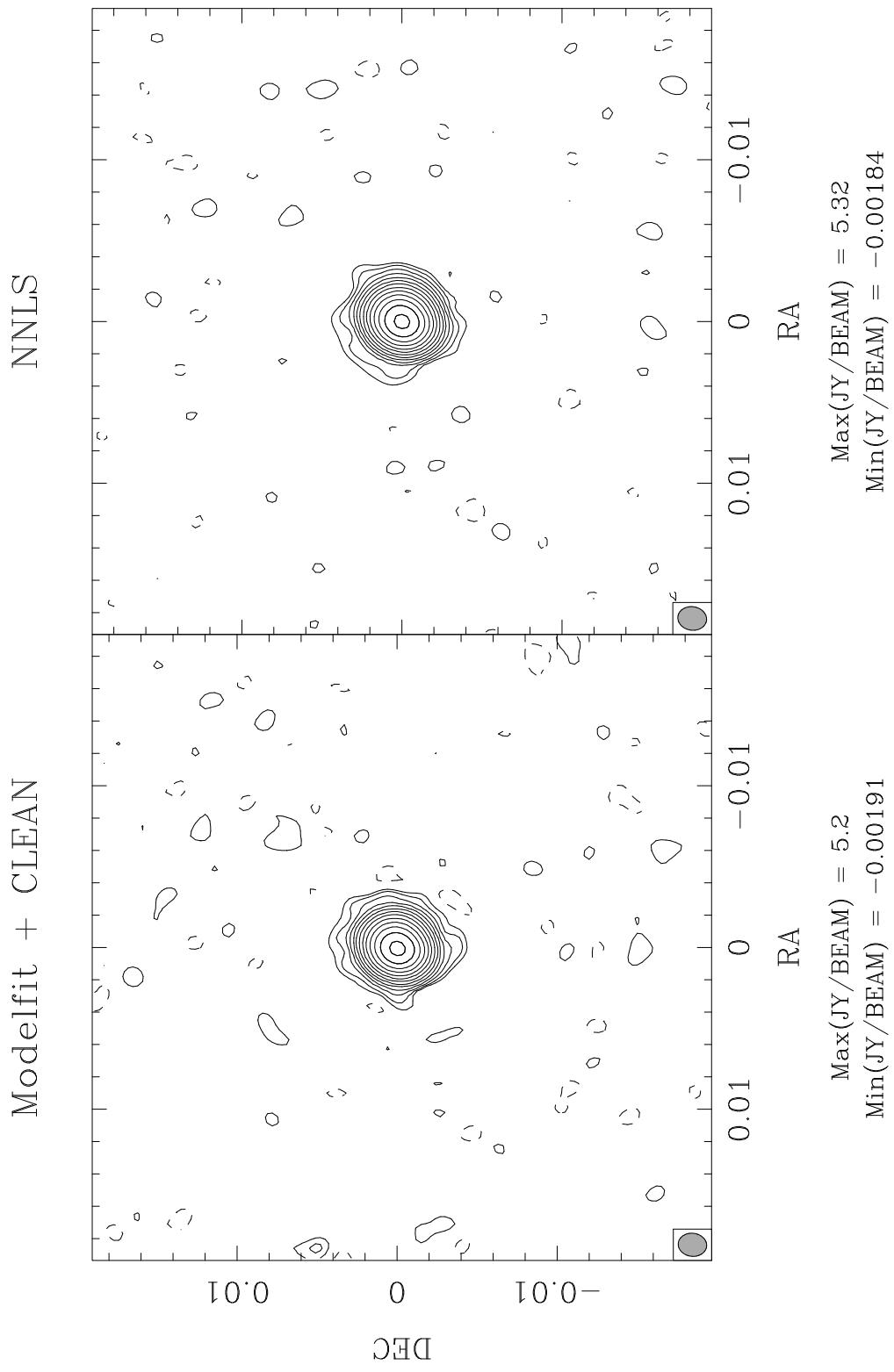


Figure 9.3: The left panel is JEC's image produced and calibrated in `difmap` via model fitting and cleaning the residual flux. The right panel is DSB's NNLS image produced and calibrated within `SDE`. Notice the extreme similarity in the (erroneous) extended structure. Dynamic range for the two images is 10,500 and 11,400 respectively.

9.1.6 NNLS

Dan Briggs has brought into operation a new deconvolver based on Non-Negative Least Squares matrix inversion. The current implementation exists in the package **SDE** as the task **svdconv**, and takes FITS format dirty map and beam as input. The algorithm is a quite straightforward application of a rather complicated preexisting constrained linear algebra algorithm. The task forms the matrix connecting an arbitrary region in the dirty map with another arbitrary region in the component plane, and solves the resulting algebraic equations with the NNLS algorithm from Lawson & Hanson (1974). The algorithm and variations will be presented in more detail in his dissertation, but the initial results are very encouraging. NNLS deconvolution is somewhat slower than existing algorithms for compact sources, and very much slower for extended objects. Memory is the limiting factor to the source size that can be deconvolved. One copy of the beam matrix must be held in memory, of size $N_{data} N_{flux}$, where N_{data} is the number of pixels in the dirty map used as input to the algorithm, and N_{flux} is the number of pixels where the algorithm is allowed to place flux. It is important that this fit into physical memory, as once page swapping sets in performance degrades dramatically, and problems that were just practical become quite impractical. Running time is roughly proportional to $N_{data} N_{flux}^2$, and also varies with the SNR of the data, with higher quality data taking longer to process. Currently a map with approximately 6000 pixels of significant emission and high SNR can be deconvolved in several hours on an IBM RS/6000.

The strength of the algorithm is obviously related to the positivity constraint, without which the problem is extremely ill posed. Since the NNLS algorithm is a direct solution, it does not suffer the problem of a spatially iterative algorithm like CLEAN, where an initial error is made at the start of the algorithm before the support constraints have a chance to become effective. This initial error produces the ring like pattern noted earlier which is never subsequently removed. As NNLS is a direct solver, the support constraints enter the solution in a globally optimal way, which appears to be quite important for self-calibration. NNLS shares the positivity constraint with the Maximum Entropy method, but unlike MEM it has no entropy (smoothness) criterion. This actually seems to be an advantage for NNLS when deconvolving compact structure. Noiseless simulations on both Gaussian and disk sources show NNLS to be much better than either MEM or CLEAN for compact objects less than a few beamwidths across. The deconvolution improvement degrades with increasing source size, but even at the computational limit of ~ 6000 pixels, it is still the best deconvolution algorithm studied.

The right panel of Figure 9.3 was made from data calibrated entirely in **SDE** using the NNLS deconvolver in the hybrid mapping loop. User input was minimal, with a single loose window around the entire region containing probable flux. The entire procedure was automated in a shell script, and the solution converged after 5 phase-only self-calibration iterations, and an additional 25 amplitude+phase iteration. Note the striking similarity of this image to that of the model fit + CLEAN procedure. These two images calibrated and

produced in completely different packages nearly convinced us that the extended emission might be real.

However, when a human entered the loop and placed a much tighter window on the allowed flux, the solution improved still further. Five more iterations of self-calibration with a tight window yields the left image in Figure 9.5. A naturally weighted map made from the same calibrated data set is also given. The extended halo is completely gone. The dynamic range is 13,700 for uniform weighting and 27,700 for natural weighting!

NNLS (UNIFORM)

NNLS (NATURAL)

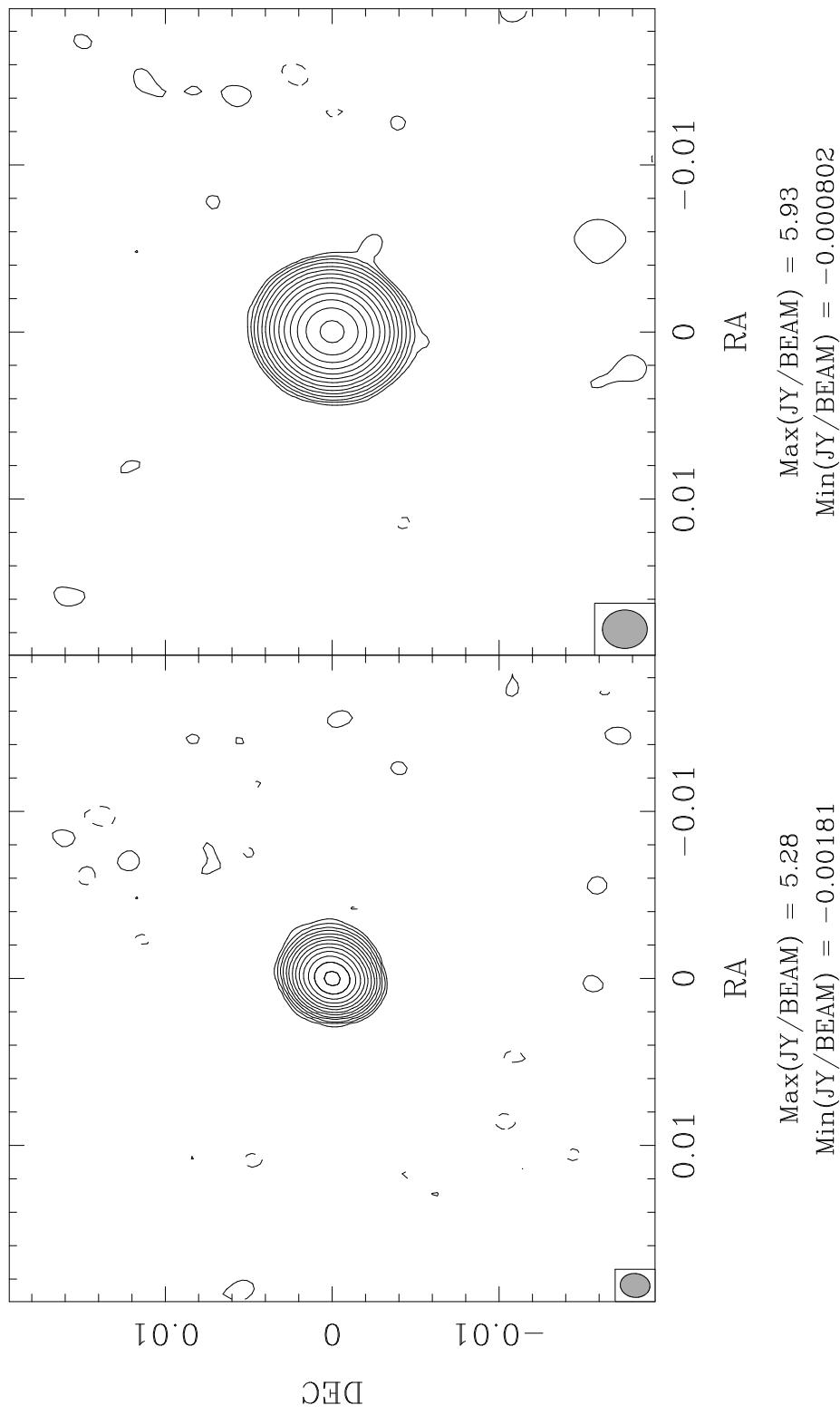


Figure 9.4: DSB's best NNLS images, with uniform and natural weighting. The difference between this figure and the right panel of Figure 9.3 is that extremely tight windows were used in the final stages of self-calibration. The naturally weighted image is contoured another factor of 2 deeper than other images presented, so that the lowest positive contour is .01% of the peak. The uniform dynamic range is 13,700 and the natural dynamic range is 27,700!

9.1.7 Noise Comparison

From the VLBA Observational Status Summary of 23-June-1994, the zenith Source Equivalent Flux Density for C band is 296 Jy. For a full track observation the effective SEFD will be degraded somewhat by elevation dependent effects, but we will use 296 to derive a best case lower limit. Equation 3 of the status summary gives the following standard expression for a single polarization naturally weighted map plane dispersion.

$$\Delta I_m = \frac{1}{\eta_S} \frac{SEFD}{\sqrt{2 \Delta\nu t_{int} N_{base}}}$$

For this observation $\Delta\nu = 1.75 \times 10^6$. The data set contains 19989 visibilities, the great majority of which are 60 second integrations. We replace $t_{int} N_{base}$ with $\Delta t N_{vis}$. The factor of $1/\eta_S$ accounts for system efficiency losses and is not yet well measured for the VLBA. In the same, “lower limit” spirit, we use $1/\eta_s = 1.57$ which is appropriate for a perfect one bit correlator at Nyquist sampling. The numbers yield an expected $\Delta I_m = .227$ mJy.

Software also exists in **SDE** to allow the estimation of thermal noise directly from the measured visibility data. The visibility averaging task can solve for the proportionality constant between the weights and inverse variance from the scatter in finite averages. After subtracting off the transformed NNLS component model, leaving mostly thermal noise, it seems to work quite well in this case. The **SDE** mapper takes proper account of weights and given this constant can calculate both the expected RMS thermal noise in the naturally weighted map and also the degradation caused by using nonnatural weights.

The averaging task produces the following proportionality constants as a function of integration time. 150 seconds: .003042. 300 seconds: .003035. 600 seconds: .003042. The point here being that it is well behaved at the one part in several hundred level. The noise behavior is close to Gaussian.

Using the 300 second figure, the expected thermal RMS noise in the maps are .259 and .508 mJy for natural and uniform weighting respectively. The measured off-source RMS from the uncleaned final NNLS images are .264 and .538 mJy.

The conclusion we draw from this is that the NNLS images are very close to the thermal RMS in terms of the noise actually present in the data. In agreement with visual inspection of the images, neither the deconvolution nor the calibration nor nonclosing errors (all of which should vary slowly on the scale of 300 seconds) are significantly contributing to the off-source RMS. In addition, the measurements are matched exactly by noise limited observations, theoretical correlator efficiency, and a mean SEFD of 340 Jy. Considering the known problems at Pie Town and Hancock, plus the effect of nonzenith observations, this is extremely good agreement between the measured thermal noise and a priori noise estimates.

9.1.8 Super Resolution

In previous super resolution work on SN1987A, (Briggs, 1994), the deconvolver of choice was MEM. At the low signal-to-noise ratio of the SN1987A data, pure deconvolution error due to incomplete sampling was completely dominated by thermal noise and the choice of algorithm was made on other criteria than high precision. MEM extrapolates well in the Fourier domain and has the property that image features are made only as sharp as mandated by the data. In fact, since agreement between the data and the model is enforced in a statistical sense only, the MEM model image¹ will always be smoother than the true image. Related to this is that the model image has an effective resolution greater than that imposed by the pixel size, and the transform of the model drops towards zero at the highest spatial frequencies. The net effect of these properties is that one can view the MEM model as an image (without residuals) that has a spatially variable resolution as high as warranted by the data. Features present in this image are guaranteed to be mandated by the data though whether by the astrophysical source or by the thermal noise fluctuations is ambiguous. Conventional astronomical usage convolves the MEM model image with a restoring beam and adds the residuals back to the image, but the MEM algorithm itself limits the maximum resolution attempted in the reconstruction, no matter what restoring beam is selected.

By contrast, while NNLS does a superb job of extrapolation in the Fourier domain near the envelope of the measured data, the spectral power does not necessarily roll off at the highest spatial frequencies. At some resolution the solution must become spurious. Direct examination of the NNLS model image in simulations shows decidedly nonphysical structure where flux in several adjacent pixels has been collapsed into one pixel of greater value surrounded by several zeros. With NNLS the onus is on the user to select a restoring beam that provides an acceptable degree of superresolution without producing obvious artifacts. Unfortunately, the restored image RMS and maximum negative as a function of restoring beam often gives little guidance as to image fidelity and a degree of arbitrary judgement must be exercised.

By itself, the CLEAN algorithm combines the worst features of the other two algorithms when used for superresolution. The high spatial frequency reconstruction is not generally as good as that of NNLS, but it requires a similarly delicate choice of optimum restoring beam. However it can be used in conjunction with MEM or model fitting. MEM suffers from the problem that even when the RMS residuals approaches the noise, there can still be considerable beam-like structure left in the residuals. Model fitting to the thermal noise limit is simply very difficult for all but the simplest sources. In both of these cases, CLEAN can be used to remove any residual structure from the residuals. Provided that

¹Actually, it is the entropy (smoothness) in the ratio between the model image and the best a priori estimate of the image that is maximized. Usually the latter is simply a constant value.

the residual structure is small, this does not destroy the superresolution properties of the solution, and the MEM + CLEAN approach can be used in situations where computational limits preclude the use of NNLS.

In Figure 9.5 we present images superresolved from the nominal 2.0×1.6 mas to 1.0×1.0 mas. These are produced from the same calibrated data as in Figure 9.4, the best NNLS calibration. The MEM algorithm was allowed to place flux in a fairly loose box around the source, though the details of the window made relatively little difference. The east-west extension of the source traces nicely the jet seen at higher frequencies. Great care must be exercised in interpreting super resolved images astrophysically, but the structure here at least seem plausible. A NNLS image superresolved to 0.6 mas was rejected for showing obvious artifacts.

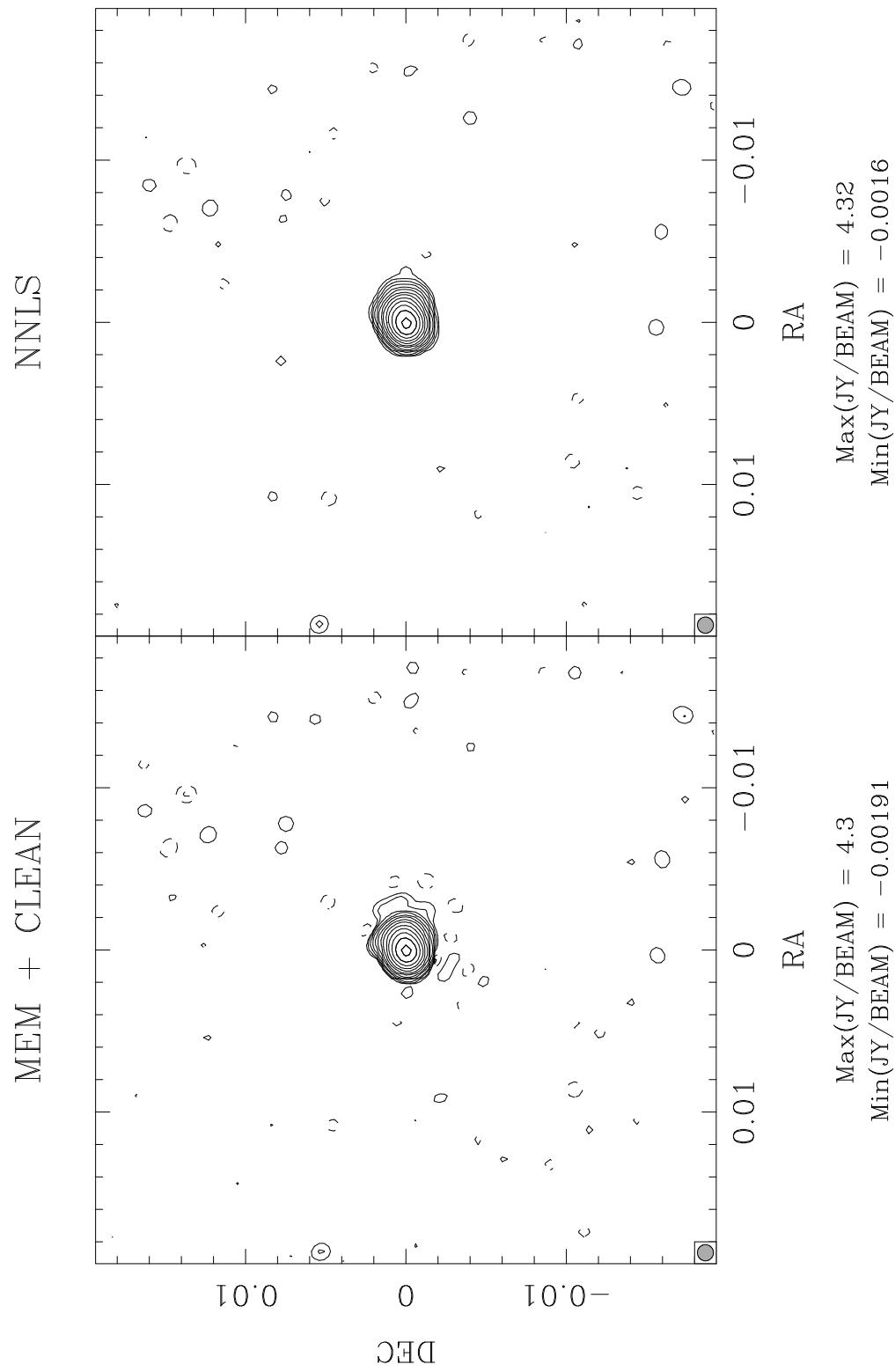


Figure 9.5: Superresolution. The left panel is produced with combination of MEM and CLEAN, the right with NNLS.

9.1.9 Self-Calibration

Clearly the interaction between the deconvolution step and self-calibration is critical in the high dynamic range limit. Both CLEAN and MEM produce much better images on the best calibrated NNLS data set than they do on data calibrated via other methods. MEM + CLEAN on the best data set produces an image comparable to NNLS, though either alone is somewhat worse.

NNLS seems to have several properties that are highly desirable in the self-calibration loop. With very little user input and loose windows it converged to a solution as good as that of the best other methods. This property allowed it to be used in an open loop shellscript where many selfcal iterations were performed automatically without user interaction. Once converged to this intermediate solution, examination of the model components provided clear delineation of the source support, and selection of a tighter flux window was straight forward. The tighter window quickly drove the iteration to the final solution.

NNLS has a clear advantage over MEM in self-calibration in that it is not biased against compact structure. NNLS has an advantage over CLEAN in that it makes efficient use of the support constraints. With CLEAN, the characteristic errors have already been introduced into the solution before the support constraints make any difference in the iteration. NNLS encodes the support information in the structure of the beam matrix and this affects the solution in a global manner.

The majority of the calibration can also be performed with lower precision deconvolvers for speed reasons. The calibration required 35 iterations to converge to the highest quality solution with NNLS, using first the loose and then the tight flux window. The calibration still required nearly the same 35 iterations to converge, restarting from the beginning and using the tight window throughout. Until the last few iterations, the intermediate solutions were nearly identical. Similar experiments varying the self calibration integration time yielded similar insensitivity to parameters until the last few iterations. The final calibration allowed independent amplitude and phase solutions for every 60 second visibility average. When the calibration was attempted with MEM as the deconvolver, intermediate solutions were very similar to that of NNLS until the MEM solution ceased to improve at about 15 iterations. Highest quality deconvolution seems necessary only at the end, and the final calibration with NNLS seems stable to different intermediate routes to the solution.

9.1.10 Conclusions and Further Work

- No obvious artifacts at all in the best images, and good agreement with a priori thermal noise. No evidence for closure errors at the 27,700:1 level! There appear to be several workable approaches to high fidelity imaging with the VLBA.

- NNLS is very promising. It is the best deconvolver when it can be applied, and it has the advantage being very forgiving about windowing. This is an excellent algorithm in the self-calibration cycle.
- Model fitting + CLEAN is a workable imaging approach and will likely drive self-calibration well.
- CLEAN & MEM individually produce image artifacts even on the best calibrated data set, but the combination is very good and not limited by source size. This does not drive self-calibration well.
- It is a bit disturbing that the first NNLS attempt & the Model fitting + CLEAN are so similar. Be conservative when interpreting such image features.
- NNLS seems to superresolve very nicely in this case – but more work would be needed before publishing it as astrophysics!
- Further testing continues on a 128 Mbits/s data set which will enable the VLBA to be tested at 150,000:1 dynamic range.

End of Memo

9.2 Simulations

Even though it appeared that the halo around DA193 could be entirely accounted for by processing affects, it was desired to verify that the VLBA could distinguish between the presence or the lack of such a halo.

Initial attempts to model the observation with statistically independent gain errors were disappointing. Rather than attempt a cure with progressively more complicated attempts to model realistic gains, I present a simple procedure to transfer the gains from the actual observation to the simulation. This makes a tremendous difference in the behavior of the simulation and yields a very powerful diagnostic technique for investigating the effects of processing on self-calibrated interferometric observations. In this case, with 3% of the total flux placed in an extended halo, it is concluded that a $\gtrsim 25,000:1$ VLBA observation can successfully distinguish unambiguously between the presence or lack of the halo.

9.2.1 Gaussian Model

The bulk of the DA193 emission is reasonably fitted with a single Gaussian component of size $.904 \times .514$ milliarcseconds, at a position angle of 109.5° . There is definitely an extended component of emission to the west that this model lacks, but it is instructive to examine whether the qualitative deconvolution behavior can be modeled with such a crude approximation to the source.

The effective thermal noise in the observations was determined by subtracting off the best deconvolved model from the visibility data set, and averaging the residual noise with the **SDE** task **visavg**. The visibilities in the observed data set were replaced by that of the Gaussian model plus the measured amount of Gaussian distributed thermal noise generated with a random number generator. The Pie Town and Hancock stations were seriously degraded in sensitivity by pointing problems and snow, respectively. In principle this degradation could be modeled properly within **SDE** by manual adjustment of the visibility weights, but for the purposes of the simulation all telescopes were assumed of equal sensitivity, and the mean thermal noise was added.

The smoothed Gaussian model and the best deconvolution of the thermally corrupted data set are given in Figure 9.6. Except where noted in the captions, all deconvolutions remaining in this chapter are done with the NNLS algorithm. Since this investigation is not concerned with super resolution and is concerned with the details of the thermal noise, the images have not had their thermal noise CLEANed to convergence as was done earlier. The same convention is followed of the lowest contour being .02% of the peak, with a factor of two between contours.

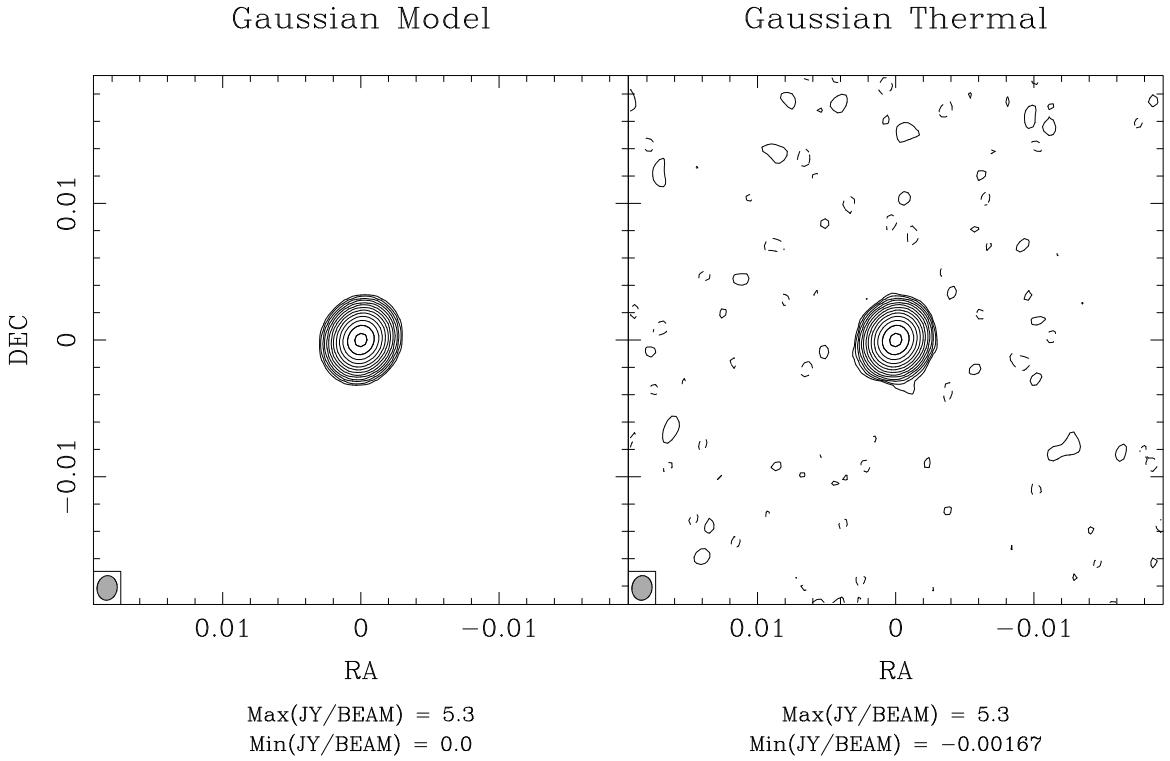


Figure 9.6: Gaussian model and best deconvolution of model + thermal noise.

9.2.2 Gain Transfer

Initial attempts to model the observation with statistically independent gain errors were disappointing. A data set was created as above, but with 2% RMS amplitude errors and completely random phases. After a single phase-only self-calibration iteration with a point source model and no time integration, the left panel of Figure 9.7 was obtained. Not only is this extremely dissimilar to the more sophisticated simulation in the right panel, but the calibration convergence of this data set is also unrealistic. Compare the off-source RMS as a function of self-calibration iteration given in Figure 9.8 for the actual data and the independent gain simulation. All self-calibration iterations were started with a point source model and 5 iterations of phase-only self-calibration. For the independent-gain simulations, the solution converged within several subsequent iterations of amplitude & phase self-calibration, as compared to an additional 30 iterations required for the real data to converge to a similar image.

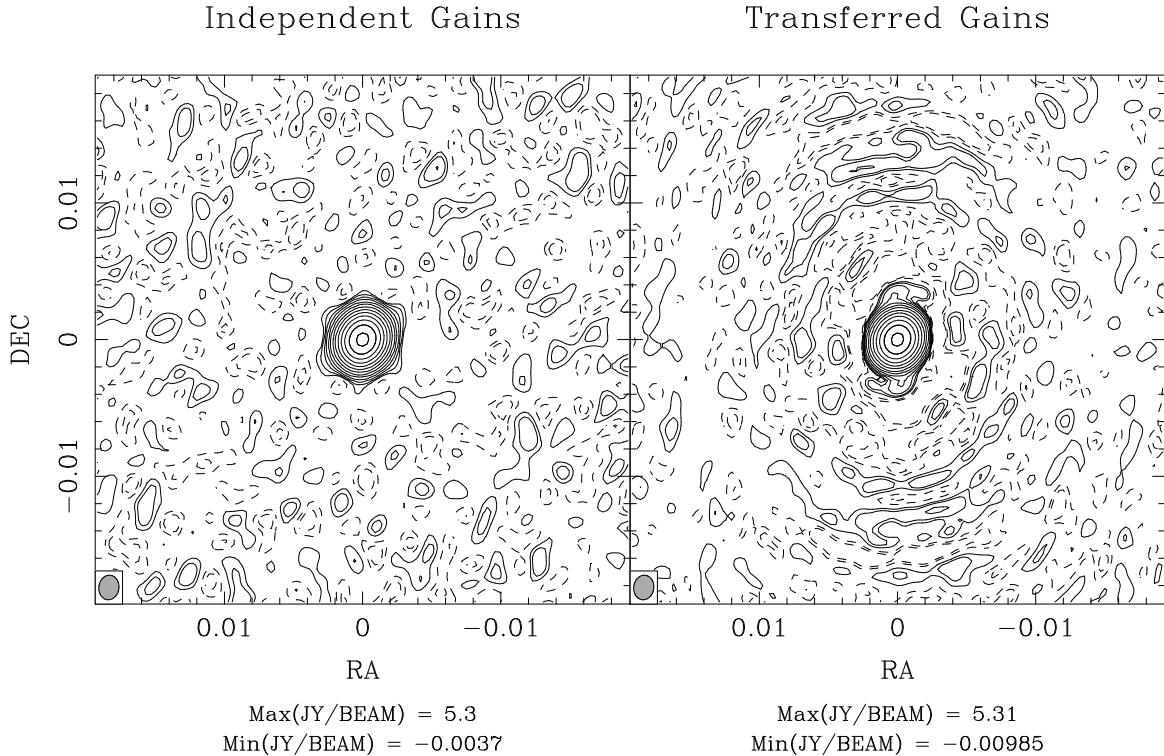


Figure 9.7: Correlated gains do make a considerable difference in simulation fidelity. The model source is a gaussian. The left panel has had greater calibration error added to it than the right panel, but without any correlation structure. After one iteration of amplitude & phase self-cal, it is already better than the more realistic simulation.

The observational image presented in the upper-right panel of Figure 9.9 used a tight deconvolution window which produces the best result. When a loose window is used in the deconvolution step, considerably fewer artifacts are seen in the independent gain

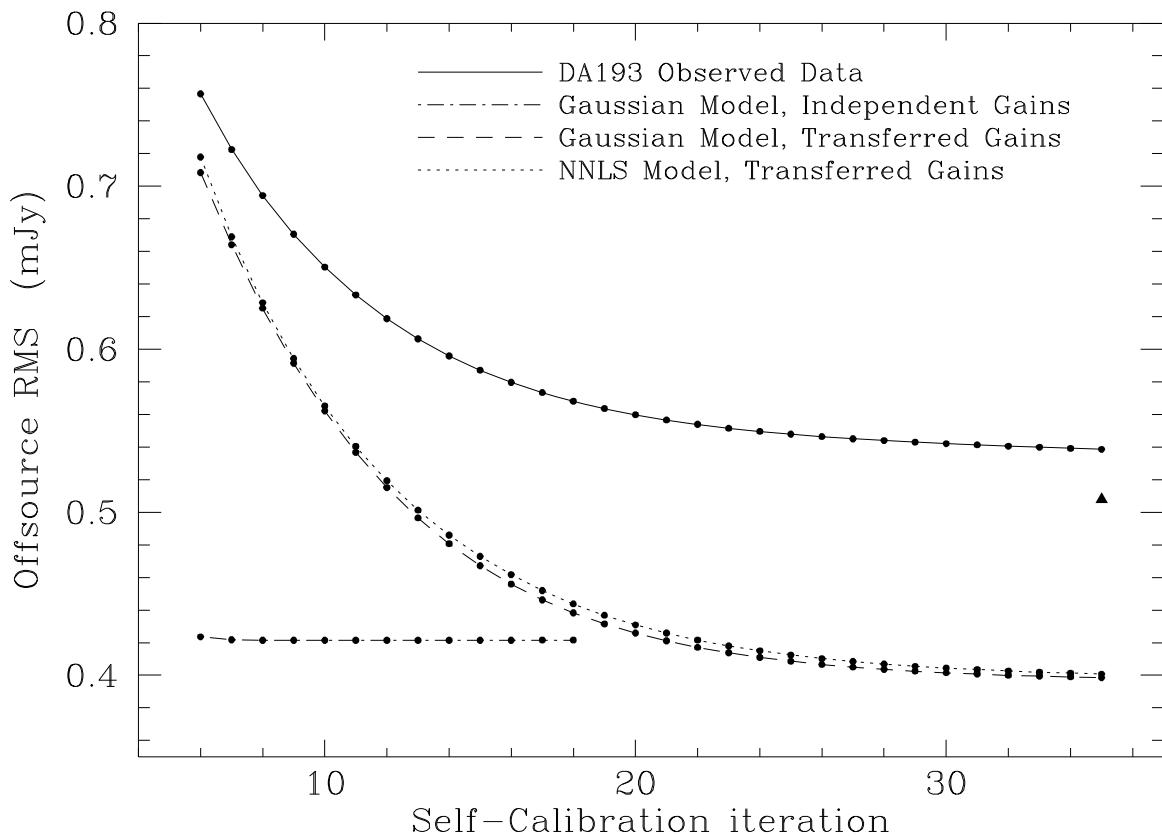


Figure 9.8: Offsource RMS against self-calibration iteration. Independent gain errors give a model that converges too fast. The black triangle is the amount of noise introduced into the model. The measured RMS is lower than that because of noise absorbed by the self calibration process.

simulation images than in either the real data or the later simulations. The correlation structure of the initial gain errors obviously makes a difference in the self-calibration cycle.

Along with several other packages, SDE has a means of simulating correlated gains. One creates a model atmosphere with given correlation structure and allows it to blow over the top of the array while observing through it. This is undoubtedly a reasonable way to proceed when the simulation is of an instrument not yet built. This procedure is complicated, however. One must select appropriate values for a number of different atmospheric parameters, many of them not well constrained by observations. One always wonders how well the model atmosphere reflects reality. In the case of a VLBI array, the assumption of a single atmosphere is clearly wrong, though it does serve to introduce a correlation structure, if perhaps not the correct one. When a well-calibrated data set of the observation under study is available, there is a better way. The visibility gains applied to the well-calibrated data set can be simply transferred to the model. In SDE this is implemented in the task **vissim** as

$$Vis'_{sim} = Vis_{sim} \times Vis_{obs}/Vis_{cal} + Noise_{therm} \quad \forall \text{ visibility samples} \quad (9.1)$$

Vis'_{sim} is the corrupted model data, Vis_{sim} the noiseless model data, Vis_{obs} the data set whose gain errors are to be transferred, Vis_{cal} is the well-calibrated data set, and $Noise_{therm}$ is Gaussian thermal noise generated with a random number generator. In this rather general formalism, the errors introduced need not be closing. Anything that has been done to the calibrated data set will be reproduced in the simulation. Of course, residual calibration errors will not be transferred, but this simple procedure is likely to produce simulations at least as accurate or better than completely synthetic data sets. Note also that the model need not resemble the observed sky at all. One could use an observation of an easily calibrated source like DA193 and apply these gains to a simulation of a much more complicated object.

The remaining two panels of Figure 9.9 show deconvolutions of the gaussian model with gains transferred from the DA193 observation. Both of these images have been produced after 35 iterations of self-calibration. The convergence of the off-source RMS to the final value in Figure 9.8 now qualitatively closely resembles that of the real data. The black triangle in that figure is the level of the thermal noise in the simulated data. (On the scale of this graph, the differences between the measured off-source RMS and the expected RMS from the thermal noise introduced into the visibility data is entirely negligible.

This simulation also illustrates another effect only recently recognized. It has long been held that the effect of self-calibration is to slightly increase the thermal noise in the map, by dint of reducing the number of good observables.² Unfortunately the iterative

²“...one should note that the thermal noise level in a self-calibrated map is slightly larger than that expected from the data statistics.” (Wilkinson, 1989, p. 89) See also Cornwell (1981).

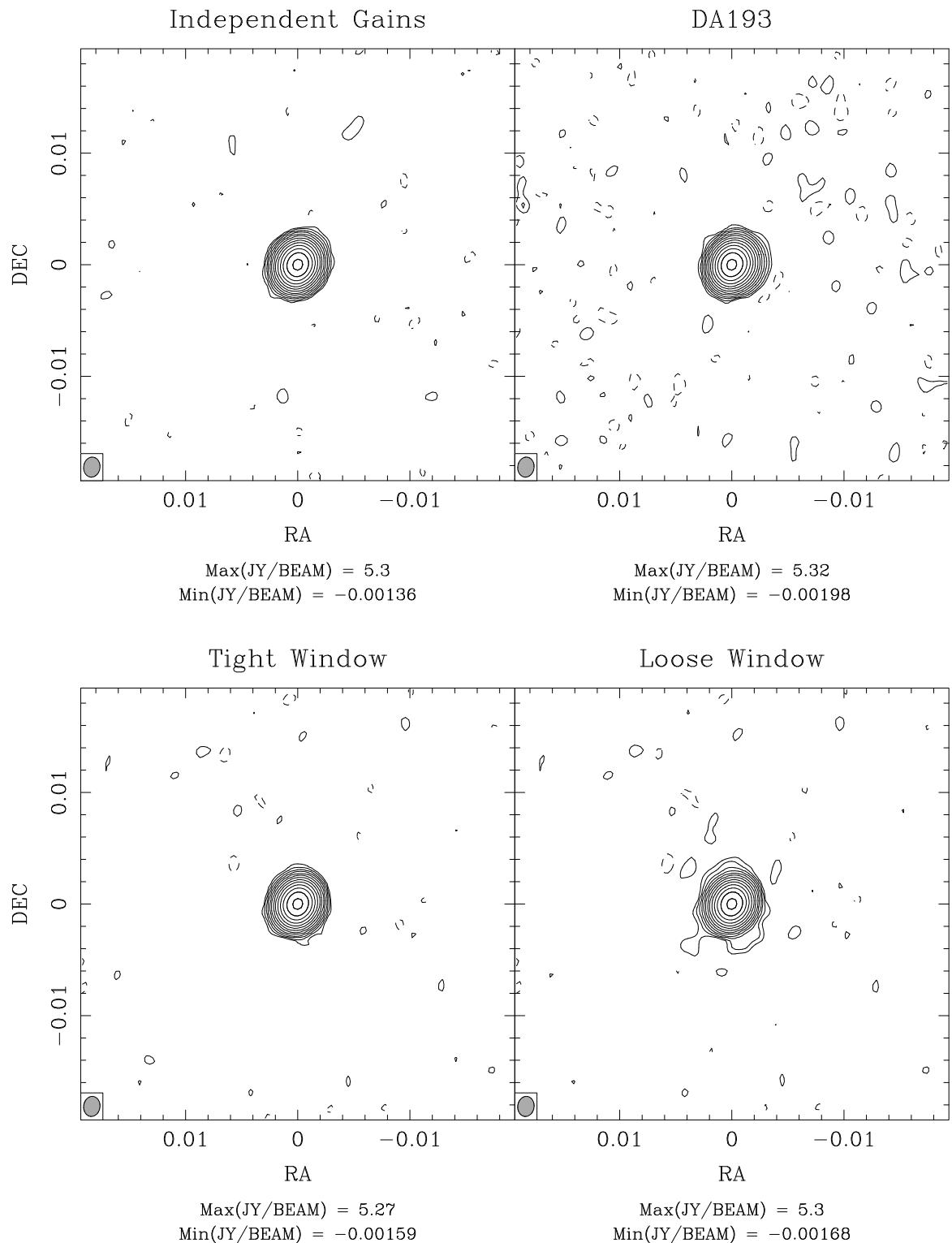


Figure 9.9: Deconvolutions of self-calibrated gaussian model data.

nature of the calibration process makes analysis of the feedback difficult. In fact, quite the opposite is true. As discovered in Wieringa's PhD dissertation work, (Wieringa, 1991), the measured off-source RMS in the map will in fact decrease by a factor of roughly

$$\sqrt{1 - N_{ant}/N_{base}}, \quad (9.2)$$

if every integration time is allowed separate calibration constants. The exact solution is not known for an arbitrary source, but the effect will be of this order. It is an apparent effect, not an actual decrease in the thermal RMS. The self-calibration algorithm has simply been told that the true sky is flat off the main source, and it has adjusted the gains so as to make this the case, introducing low level calibration errors in the process. This is difficult to avoid, but the effect should be accounted for in the highest quality simulations. When measuring noise from self-calibrated data, one should either correct by the inverse of equation 9.2 to find the appropriate amount of noise to introduce into the uncalibrated data to match observation, or one should simply correct by the ratio of expected-to-measured thermal noise in a first pass through the simulations. In this memo, the difference of approximately 20% in RMS noise level is not thought to affect the final conclusions, but if desired another round of simulations could be done which would presumably produce an RMS convergence curve quantitatively very similar to the measured trace of Figure 9.8. In Figure 9.10, we show a model observation using an NNLS model which closely agrees with the observational data after several iterations of self-calibration.

9.2.3 Halo Model

The central question of this investigation is: *Can we create a halo in the processing if is none is present in the data?* and the related *If the halo is present in the data, will we see it?*

Three percent of the source flux was removed from the single-Gaussian model and placed in an azimuthally-symmetric halo. The model, smoothed down to the resolution of the restoring beam, is shown in the upper-left panel of Figure 9.11. This model is intended to qualitatively represent the halo seen in early processing stages of the DA193 data, shown in Figure 9.2. The upper-right panel shows how well the source can be deconvolved with a careful selection of window, in this case a circle of radius 4.44 mas, centered on the source. CLEAN, allowed a somewhat larger window of radius 6.00 mas, produced the figure on the lower-left. Finally, if NNLS is boxed very tightly to the support of the central Gaussian, the figure on the lower-right results. In all of these images, the halo is clearly visible. While the mis-boxed NNLS deconvolution is not so pretty as the others, no one could fail to miss that there is something wrong.

Now, how well can we hide or find the halo when we include self-calibration in the loop? The next set of images were self calibrated for the same 35 iterations as were the other images. In the upper-left panel the same over-tight window was used in an effort to make

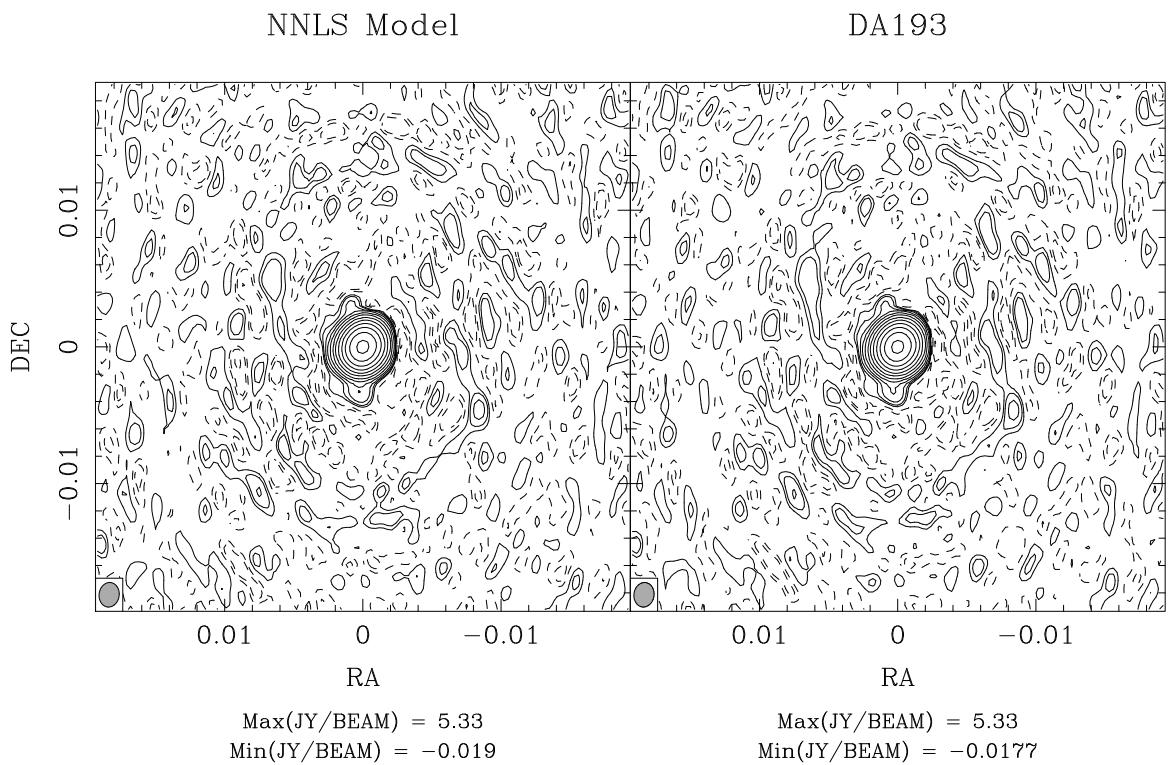


Figure 9.10: Correlated gain example with an NNLS model and transferred gains. The right panel is a simulated data set after 5 phase only self-calibrations and 1 iteration of amplitude and phase self-calibration. The right panel is the observational data, and they track each other quite well.

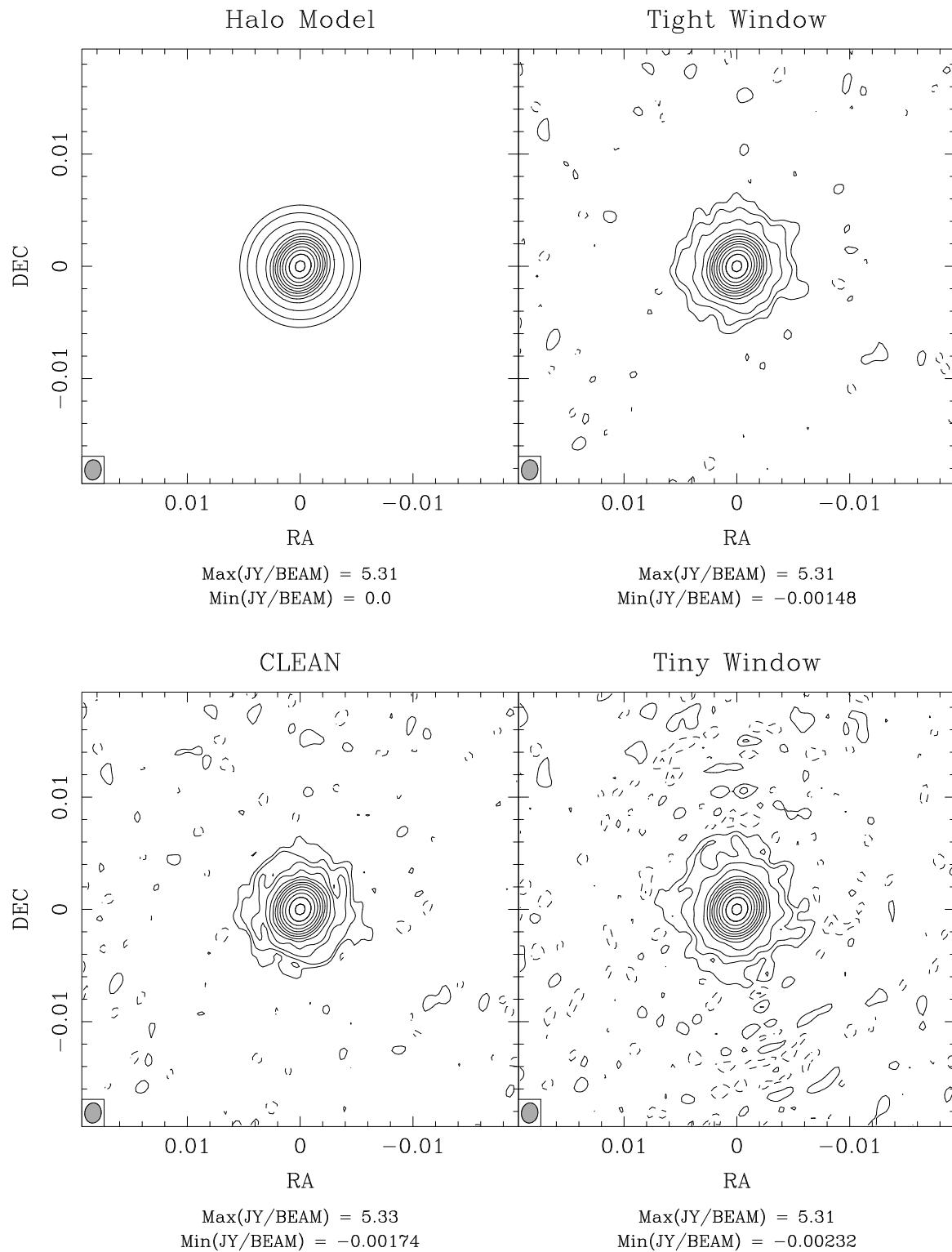


Figure 9.11: Halo model and best deconvolution of model + thermal noise.

the halo vanish. It is certainly less obvious than in the previous image, but I think there would be little chance of missing its existence here, either. The upper-right image is what results from iteration on a large window of radius 11.4 mas. The quality of reconstruction is not as pretty as before, but the halo is most certainly there. The final point to make is that if one starts with these two images as a model, then resets the flux window to the optimum region used in Figure 9.11, the solution immediate converges within several additional iterations to the lower two figures. Whether one starts too tight and expands the window or starts too loose and contracts the window makes little difference. The solution is fairly robust about the path taken to find it.

If a halo was present at the 3% or 1000:1 level in the first observation described, we would have seen it. The confidence in the imaging is high.

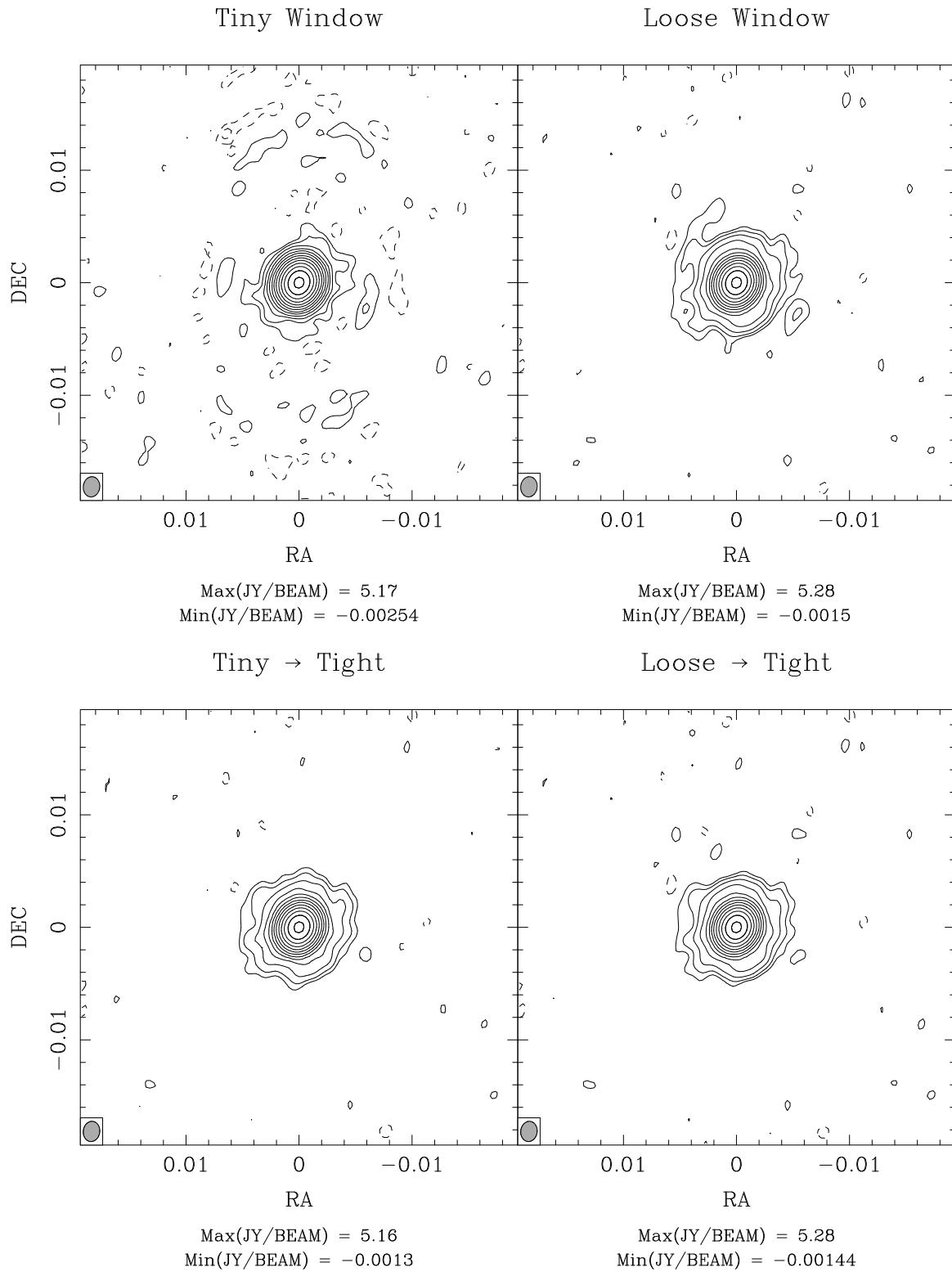
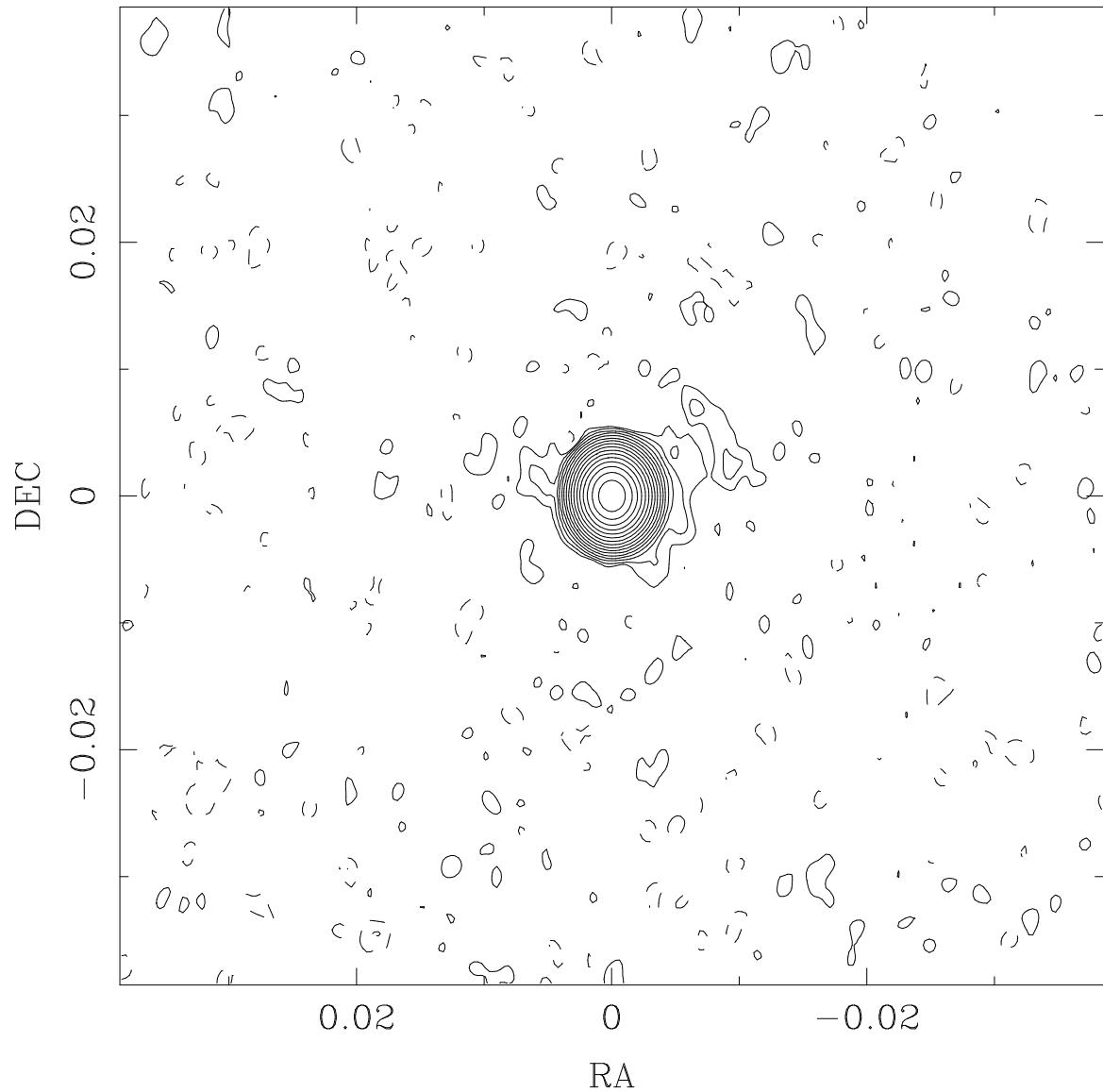


Figure 9.12: Deconvolutions of self-calibrated halo model data.

9.3 Conclusions and Further Work

As mentioned in the introduction to the first memo, further testing of the VLBA is indeed underway. In a gratifying confirmation of the previous conclusion, Tim Cornwell has been working on a data set with nearly the full sensitivity of the VLBA. His observations consist of 8 8-MHz intermediate frequency channels for a total of 64 MHz bandwidth. Using clock offsets determined from a previous correlation of the data set, he has been able to produce what appears to be nearly a noise-limited observation at a dynamic range of 115,000:1. Ironically, we are now in a similar position to when we started, wondering if the diffuse emission we see around the source might be real. But now we are wondering about emission a full order of magnitude fainter. At this level, instrumental effects are subtle, to say the least, and it will be challenging to properly asses the fidelity level of observations like these. But there is not doubt that we are better prepared to do so than we were several years ago.

DA193 C band (8 IF)



Max(JY/BEAM) = 5.83
Min(JY/BEAM) = -0.00021

Figure 9.13: High dynamic range image of DA193. Peak to off-source RMS is 115,000:1!

Chapter 10

Concluding Comments

This dissertation has largely been driven by a fresh look at several old problems and established practices. The fresh look paid off here, and there is probably much material in a similar vein waiting to be discovered or improved. Uniform and natural weighting have been around as long as interferometric imaging, and yet a simple alternative was found that is demonstrably superior for some applications. Just because something has been done one way for a long time doesn't mean that it can't be improved.

We have seen that a simple modification to uniform weighting leads to a smoothly varying intermediate form between uniform and natural weighting. Different values of the new weighting parameter, the robustness, can produce images at different points along the resolution-thermal noise tradeoff curve. One can obtain moderately improved thermal noise characteristics compared to uniform weighting at little cost in resolution, or higher resolution compared to natural weighting at little cost in thermal noise limited sensitivity. The near-uniform regime will be particularly valuable to VLBI arrays with elements of greatly differing individual sensitivities. The near-natural regime will be particularly valuable to arrays like the VLA which have a preponderance of short spacings; the new weighting often produces extremely low sidelobes and a particularly good match between the dirty beam and its fitted Gaussian. It is an excellent choice for imaging extended emission.

In addition to robust weighting, several other improvements to existing weighting practice were given, namely generalized tapering and gridless uniform weighting. My suspicion is these new techniques, plus proper use of the tools that we already have, have entered the region of diminishing returns in terms of conventional weighting by optimizing a single criterion independently of the source. (The global minimization of individual weights to best fit the PSF to a Gaussian over a finite window is a possible exception.) For the most part, I consider this project taken to a logical plateau. It may not be finished, but I think the next significant step will be another new concept rather than an incremental improvement over the ideas presented here. Quite possibly it will come from directly involving a model of the source structure in the weighting process, and it may even involve abandoning the shift invariant point spread function formalism all together. The latter might involve a different weighting on a per pixel basis to move far sidelobes around, or it might involve a deconvolution algorithm that operates strictly in the visibility plane, such as given by Wilczek and Drapatz (1985).

By contrast with visibility weighting, algebraic deconvolution is an area where much work remains to be done. The general strategy of limiting the source size to as small as necessary and deconvolving directly instead of iteratively has been shown quite successful for some sources. Still, the constraints imposed by the existing algorithms are uncomfortably tight. NNLS needs to be extended to larger sources, both from a computational efficiency standpoint and from a quality of solution standpoint. An obvious attack is to enforce a correlation between pixels with a non-delta-function pixel model. This has already been shown to be effective in improving the solution on extended sources, but exactly how effective, and how much one is hurt on compact sources is not clear, nor is the proper selection of pixel function. This is an active research topic at the moment.

The particular nature of the Lawson-Hanson NNLS algorithm needs to be explored. The algorithm is still selecting a solution from the invisible distribution, albeit a much more constrained one. NNLS deconvolution should be implemented with other procedures which solve the equivalent problem. Is the quality of the solution better, worse, or the same? That is, are the favorable qualities of the deconvolver seen here a generic consequence of the information being added to the problem, or a particular consequence of the manner in which this algorithm operates.

The great flexibility of the linear algebraic deconvolution approach needs to be exploited. Having gone to all the trouble of storing separately each and every redundant pixel in the beam matrix, it seems a shame not to make use of a more general weighting scheme than that of a simple convolution equation. Perhaps a space variant pixel function should be considered? This line of thinking begins to resemble the pixon approach of Puetter and Piña, (1994), though the exact correspondence is not clear. Granted, the case where a space variant point spread function is forced upon us by geometrical considerations, is precisely the large image case that the algebraic methods cannot handle. But there may well be use for deliberately deforming the convolution equation into something more amenable to high-precision solution.

I believe that there is still considerable room for improvement left in the representation area, in general. This dissertation has shown exactly how the representation of the convolution equation may be extended with the use of an arbitrary pixel function. Similar work is given in Conley et al. (1994), where the pixel function is restricted to $\Pi(x/\Delta x)\Pi(y/\Delta y)$, so presumably other authors are also unhappy with the current representation methodology. I have long been intrigued by the contrast between the finite difference and finite element methodologies of representing partial differential equations, (Lapidus and Pinder, 1982). The standard discretization resembles the finite difference approach in that it represents the variable of interest at a particular point, namely the center of the pixels. Finite volume, (a particular case of the finite element methodology), instead represents a pixel by the total flux contained within it, rather than a sample at a particular point. The differential equation to be solved is then posed as a series of boundary conditions between pixels, rather than point equations to be solved at the pixel centers with the

approximated derivatives. Might we not do something similar in posing the deconvolution equation as a series of boundary equations on some suitable function of the sky brightness? An integral equation is not a differential equation, and the direct relationship of the boundary conditions to the fundamental equation to be solved not so apparent. But the problem need not be posed in terms of the sky brightness itself — one could work in the space of the derivative or cumulative integral of the sky, for instance. In a suitable space, a boundary condition formalism might be appropriate and productive.

Another very practical topic for future research involves calibration. The slow convergence of self-calibration seen in the DA193 examples suggests that it may be productive to integrate the deconvolution and the calibration step into one larger procedure that has knowledge of both, so that we may take a step in the direction of the true gradient instead of merely alternating the steps in the orthogonal subspaces. Succeeding combined deconvolution and self-calibration steps could use a proper conjugate gradient approach, and this could potentially reduce the number of total iterations considerably.

Finally, I think the considerable practical utility of modelling in diagnosing deconvolution related problems has been well demonstrated here. I believe that modelling needs to be a more routine part of all deconvolutions where it is feasible. In work not presented, it was attempted to write a point & click style Gaussian model-fitting program. Initial guesses to the model parameters were generated semi-automatically, with some input being given by the user and others being calculated. Even with a sophisticated external program to do the hard work of the minimization, this proved a surprisingly hard problem. Even so, this is a problem that badly needs to be solved, and needs to become available in the standard analysis packages. As it stands now, model fitting to all but the simplest images is a fairly tedious process, and people are reluctant to do so unless forced by the needs of their particular project. Modelling must be made sufficiently simple that people will *use* it as a routine part of their data analysis.

Deconvolution and image-processing are as much a part of modern interferometric imaging as the telescope, yet there is no substitute for a stable and well calibrated telescope. Ultimately, if one wishes to know the source visibility function at a given u - v point, the most reliable method is to measure it. But equally true is that given an observation of some fixed quality and completeness, we should strive to extract the highest possible quality images and scientific results from it.

Appendix A

Lagrange Multipliers and the Weighted Mean

Extremum problems with equality constraints are often solved with Lagrange multipliers. An unconstrained extremum problem merely sets the gradient of the objective function to zero. If there additional equality constraints to be satisfied, the problem can be transformed into an equivalent unconstrained extremum problem by the introduction of an additional term into the objective function proportional to an unknown scalar, the Lagrange multiplier. With a single constraint to be satisfied, the statement of the method is particularly simple.

If a scalar field $f(x_1, \dots, x_n)$ has a relative extremum when subject to the constraint $g(x_1, \dots, x_n) = 0$, then there exists a scalar λ such that $\nabla f - \lambda \nabla g = 0$.

In its full generality, the method is fairly difficult to derive. However, nearly any advanced calculus text will provide sufficient information to guide the use of the method safely. See, for example, Apostol (1969). For our purposes, the application of the method is quite straightforward. Consider the problem of the weighted mean:

Given N independent random variables A_i with common mean μ , find a linear sum A with expectation μ and minimal variance.

First, note that

$$\begin{aligned}\langle A \rangle &= \langle \sum_i w_i A_i \rangle = \sum_i w_i \langle A_i \rangle = \mu \sum_i w_i \equiv \mu \\ \Rightarrow \quad \sum_i w_i &= 1\end{aligned}$$

Of course, if μ is zero, then the sum of the weights is not constrained. In this case, we must choose the weight sum to be unity as part of the problems statement.

$$\sigma_A^2 = \langle A^2 \rangle - \langle A \rangle^2 = \langle (\sum_i w_i A_i)^2 \rangle - \mu^2 \tag{A.1}$$

To minimize σ_A^2 subject to $\sum w_i = 1$, we minimize $Q \equiv \sigma_A^2 + \lambda(\sum w_i - 1)$, where λ is an unknown constant.

$$\frac{\partial Q}{\partial w_k} = \frac{\partial}{\partial w_k} \left[\langle (\sum_i w_i A_i)^2 \rangle - \mu^2 - \lambda(\sum_i w_i - 1) \right]$$

$$\begin{aligned}
&= \left\langle 2\left(\sum_i w_i A_i\right) \frac{\partial}{\partial w_k} \left(\sum_i w_i A_i\right) \right\rangle - \lambda \\
&= \left\langle 2\left(\sum_i w_i A_i\right) A_k - \lambda \right\rangle \\
&= \left\langle 2 \sum_{i \neq k} w_i A_i A_k + 2w_k A_k^2 - \lambda \right\rangle \\
&= 2 \sum_{i \neq k} w_i \langle A_i \rangle \langle A_k \rangle + 2w_k (\mu^2 + \sigma_k^2) - \lambda \\
&= 2\mu^2 \sum_i w_i + 2w_k \sigma_k^2 - \lambda = 0 \\
\Rightarrow & w_k \sigma_k^2 = \lambda/2 - \mu^2 = \text{const} \\
\Rightarrow & w_k \propto 1/\sigma_k^2
\end{aligned}$$

The normalization of the weights results in the conventional expression

$$w_k = \frac{1}{\sigma_k^2} \left/ \sum_i \frac{1}{\sigma_i^2} \right. \quad (\text{A.2})$$

Note that in contrast to the usual derivation of this result, no assumptions at all are needed about the random variables A_i beyond their having a finite variance. In particular, they are not assumed to be Gaussian.

Appendix B

Elliptical Gaussian Convolution

Elliptical Gaussians are used frequently in radio astronomical imaging package such *AIPS*, *SDE*, and the Caltech VLBI package. Unfortunately, they are tedious to deal with analytically, and as a consequence some programs which should support full elliptical (inverse) tapering and/or convolution, do not. I present here the equations and algorithms necessary to deal with nearly arbitrary tapers and convolutions as reference and encouragement to authors of future programs. The derivation is somewhat similar to, but more general than Wild (1970).

The representation of tapers and convolutions by signed quadratic coefficients leads to a very flexible system, and is probably among the simpler approaches given the complexity of the problem. The primary disadvantage is that this representation cannot handle one-dimensional convolutions. This would require an axis of the beam to be zero and correspondingly infinite coefficients. Recasting the formalism to work in terms of the beam axes instead of their inverses is not sufficient to remove the singularity. Relaxing the restriction on one-dimensional convolutions would require special purpose code that completely bypasses this formalism. Fortunately, most practical problems can be solved with a highly elongated two-dimensional convolution, so the necessity for implementing this special case is primarily aesthetic.

The elliptical Gaussians are most often described by the major and minor axis full width at half maximum¹. The position angle is that of the major axis measured from the north through east, or counter clockwise from vertical under the normal mapping conventions. The unrotated ellipse of half maximum is

$$\frac{x^2}{(B_{maj}/2)^2} + \frac{y^2}{(B_{min}/2)^2} = 1 \quad (\text{B.1})$$

and the normalization of the gaussian exponent is fixed by

$$e^{-k} = 1/2 \quad \Rightarrow \quad k = \ln 2.$$

Defining $A_0 \equiv 1/(B_{maj}/2)^2$, $C_0 \equiv 1/(B_{min}/2)^2$ and rotating by an angle $\theta \equiv B_{pa} + 90^\circ$, we can write the elliptical gaussian as

$$g(x, y) = \exp(-Ax^2 - Bxy - Cy^2), \quad \text{with}$$

¹Other choices are described in (Sramek and Schwab, 1989, p. 121).

$$\begin{aligned}
A &= (A_0 \cos^2 \theta + C_0 \sin^2 \theta) \ln 2 \\
B &= (-(C_0 - A_0) \sin(2\theta)) \ln 2 \\
C &= (A_0 \sin^2 \theta + C_0 \cos^2 \theta) \ln 2
\end{aligned} \tag{B.2}$$

From the basic Fourier transform pair

$$e^{-Ax^2} \stackrel{\mathcal{F}}{=} \sqrt{\pi/A} e^{-\pi^2 u^2 / A}$$

We can work out the transform of

$$\begin{aligned}
\mathfrak{F} [e^{-Ax^2 - Bx}] &= \mathfrak{F} [e^{-(Ax^2 + Bx + B^2/4A)} e^{B^2/4A}] \\
&= e^{B^2/4A} \mathfrak{F} [e^{-A(x+B/2A)^2}] \\
&= e^{B^2/4A} e^{-i2\pi(-B/2A)u} \mathfrak{F} [e^{-Ax^2}] \\
&= \sqrt{\pi/A} e^{-\pi^2 u^2 / A + B^2/4A + i\pi Bu/A}
\end{aligned}$$

This in turn allows us to work out the transform of $g(x, y)$

$$\begin{aligned}
\mathfrak{F}_2 [g(x, y)] &= \mathfrak{F}_y \mathfrak{F}_x [g(x, y)] \\
&= \mathfrak{F}_y \mathfrak{F}_x [\exp(-Ax^2 - By x) \exp(-Cy^2)] \\
&= \mathfrak{F}_y [\exp(-Cy^2) (\pi/A)^{1/2} \exp(B^2 y^2 / 4A) \exp(i\pi Bu/A) \exp(-\pi^2 u^2 / A)] \\
&= \mathfrak{F}_y [\exp[-(C - B^2/4A)y^2 + (i\pi Bu/A)y] (\pi/A)^{1/2} \exp(-\pi^2 u^2 / A)] \\
&= \left(\frac{\pi}{C - B^2/4A} \right)^{1/2} \exp \left(\frac{(i\pi Bu/A)^2}{4(C - B^2/4A)} \right) \exp \left(\frac{i\pi(-i\pi Bu/A)v}{C - B^2/4A} \right) \\
&\quad \times \exp \left(\frac{-\pi^2 v^2}{C - B^2/4A} \right) \left(\frac{\pi}{A} \right)^{1/2} \exp \left(\frac{-\pi^2 u^2}{A} \right) \\
&= 2\pi(4AC - B^2)^{-1/2} \\
&\quad \times \exp \left(\frac{\pi^2}{4AC - B^2} (-B^2 u^2 / A + Buv + (B^2 - 4AC)u^2 / A - v^2 / 4A) \right) \\
&= 2\pi(4AC - B^2)^{-1/2} \exp \left(\frac{4\pi^2}{4AC - B^2} (-Cu^2 + Buv - Av^2) \right)
\end{aligned} \tag{B.3}$$

Thus prepared, we can now calculate the transform of two convolved Gaussians

$$\begin{aligned}
g_1(x, y) &\equiv \exp(-A_1 x^2 - B_1 xy - C_1 y^2) \\
g_2(x, y) &\equiv \exp(-A_2 x^2 - B_2 xy - C_2 y^2)
\end{aligned}$$

$$\begin{aligned}
\mathfrak{F}_2 [g_1 * g_2] &= \mathfrak{F}_2 [g_1] \mathfrak{F}_2 [g_2] \\
&\propto \exp \left(-4\pi^2 \left(\left(\frac{C_1}{4A_1 C_1 - B_1^2} + \frac{C_2}{4A_2 C_2 - B_2^2} \right) u^2 + \dots \right) \right)
\end{aligned}$$

We can see by inspection that the resulting exponential will be a quadratic form in u and v . Hence we write the following identifications²

$$\begin{aligned}\frac{A_{out}}{A_{out}C_{out} - B_{out}^2} &= \frac{A_1}{4A_1C_1 - B_1^2} + \frac{A_2}{4A_2C_2 - B_2^2} \equiv R_A \\ \frac{B_{out}}{A_{out}C_{out} - B_{out}^2} &= \frac{B_1}{4A_1C_1 - B_1^2} + \frac{B_2}{4A_2C_2 - B_2^2} \equiv R_B \\ \frac{C_{out}}{A_{out}C_{out} - B_{out}^2} &= \frac{C_1}{4A_1C_1 - B_1^2} + \frac{C_2}{4A_2C_2 - B_2^2} \equiv R_C\end{aligned}\quad (\text{B.4})$$

B might be zero, but both A and C should be nonzero. One can solve for A_{out} , B_{out} and C_{out} via

$$\begin{aligned}C_{out} &= A_{out}R_C/R_A \quad B_{out} = A_{out}R_B/R_A \\ \frac{A_{out}}{4A_{out}^2R_C/R_A - A_{out}^2R_B^2/R_A^2} &= R_A \quad \Rightarrow \quad A_{out} = \frac{1}{4R_C - R_B^2/R_A}\end{aligned}\quad (\text{B.5})$$

The solution for A_{out} in equation B.5 can be rather ill-conditioned. In a computer implementation one might solve for both A_{out} and C_{out} in this way, and select the solution path with the larger denominator. These equations can be solved as easily for one input as for the output, so they can be used for deconvolution as well as convolution. Converting the resulting coefficients back to elliptical parameters is straightforward analytic geometry. In program B.1 we present a general algorithm to perform the conversion which can be used for this and other purposes. Here, we simply note that after calling this routine with $D = E = 0$ and $F = -1/\ln 2$, the conventional beam parameters of the convolution result are $B_{maj} = 2S_{maj}$, $B_{min} = 2S_{min}$ and $B_{pa} = \phi 180^\circ/\pi + 90^\circ$. The position angle is arbitrary to multiples of 180° , though it is probably wise to constrain it to within $\pm 90^\circ$. Since convolutions are linear, the amplitude can be handled separately, using the fact that

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp \left(\frac{-\ln 2}{(B_{maj}/2)^2} x^2 + \frac{-\ln 2}{(B_{min}/2)^2} y^2 \right) dx dy = \frac{\pi}{4 \ln 2} B_{maj} B_{min} \approx 1.1331 B_{maj} B_{min}$$

The conversion factor between flux density per pixel and per beam is this quantity over $\Delta\ell \Delta m$.

²If the equations are recast in terms of B_{maj} and B_{min} , R_A and R_C remain finite even in the zero limit. Sadly, R_B does not and the resulting equations in R_A, R_B & R_C are quite difficult to solve even when treating the limits as special cases.

```

begin
  if  $B^2 - 4AC = 0 \rightarrow$  error;
  if  $A \neq C$ 
     $\phi := \tan^{-1}(B/(A - C))$ 
    if  $\phi < 0 \rightarrow \phi := \phi + \pi$ 
     $\phi := \phi/2$ 
  else
     $\phi := \pi/4$ 
   $A_1 := A \cos^2 \phi + B \cos \phi \sin \phi + C \sin^2 \phi$ 
   $B_1 := 2(C - A) \sin \phi \cos \phi + B(\cos^2 \phi - \sin^2 \phi)$ 
   $C_1 := A \sin^2 \phi - B \cos \phi \sin \phi + C \cos^2 \phi$ 
   $D_1 := D \cos \phi + E \sin \phi$ 
   $E_1 := -D \sin \phi + E \cos \phi$ 
   $x_{c1} := -D_1/2A_1$ 
   $y_{c1} := -E_1/2C_1$ 
   $x_c := x_{c1} \cos \phi - y_{c1} \sin \phi$ 
   $y_c := x_{c1} \sin \phi + y_{c1} \cos \phi$ 
  if  $A_1 = 0 \vee C_1 = 0 \rightarrow$  error;
   $A_2 := (-F + D_1^2/4A_1 + E_1^2/4C_1)/A_1$ 
   $C_2 := (-F + D_1^2/4A_1 + E_1^2/4C_1)/C_1$ 
   $S_{maj} = \text{sgn}(A_2)\sqrt{|A_2|}$ 
   $S_{min} = \text{sgn}(C_2)\sqrt{|C_2|}$ 
  if  $|A_2| < |C_2|$ 
     $S_{maj} := \text{sgn}(C_2)\sqrt{|C_2|}$ 
     $S_{min} := \text{sgn}(A_2)\sqrt{|A_2|}$ 
     $\phi := \phi + \pi/4$ 
end

```

Input form is $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$
Form is a parabola
Find rotation angle that zeros crossterm
All angles in radians
Degenerate case – assign a value
Transform to rotated coords
 B_1 should be zero
Find center by completing the square
and rotating back to original coordinates
Ellipse/Hyperbola is degenerate
Convert to form $x^2/A_2 + y^2/C_2 = 1$
 $S_{maj}S_{min} > 0 \Rightarrow$ Elliptical
 $S_{maj}S_{min} < 0 \Rightarrow$ Hyperbolic
Ensure $S_{maj} \geq S_{min}$
Results are $x_c, y_c, S_{maj}, S_{min}, \phi$

Program B.1: Conversion of quadratic parameters to elliptical/hyperbolic parameters

Appendix C

Beam Fitting

In the same spirit as Appendix B, we present an explicit derivation of the Gaussian beam fitting equations. The problem can be linearized by taking logarithms. This yields an analytic formulation that is both fairly robust and amenable to direct solution. If one assumes that beams formed via Fourier synthesis are normalized to a peak of unity and centered on a reference pixel which we shift to the origin, the function to be fit is simply

$$\begin{aligned} G(x, y) &\equiv \exp(-Ax^2 - Bxy - Cy^2) \\ -\ln G &= Ax^2 + Bxy + Cy^2. \end{aligned}$$

This is a linear problem for the three constants A , B & C , with the basis functions x^2 , xy & y^2 . The normal equations as described in section 14.3 of Press et al. (1986), become

$$\begin{bmatrix} \sum x^4 & \sum x^3y & \sum x^2y^2 \\ \sum x^3y & \sum x^2y^2 & \sum xy^3 \\ \sum x^2y^2 & \sum xy^3 & \sum y^4 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \sum -x^2 \ln G \\ \sum -xy \ln G \\ \sum -y^2 \ln G \end{bmatrix} \quad (\text{C.1})$$

This set of equations can be solved in usual way, but it is particularly useful to call the previous matrix $\boldsymbol{\alpha}$, and define

$$T \equiv \alpha_{11}\alpha_{22}\alpha_{33} - \alpha_{11}\alpha_{23}^2 - \alpha_{22}\alpha_{13}^2 - \alpha_{33}\alpha_{12}^2 + 2\alpha_{12}\alpha_{13}\alpha_{23}$$

Call the righthand side of equation C.1, $\boldsymbol{\beta}$. If T is larger than a few times the machine precision, we can write the quadratic coefficients as

$$\begin{aligned} A &= ((\alpha_{22}\alpha_{33} - \alpha_{23}^2)\beta_1 + (\alpha_{13}\alpha_{23} - \alpha_{12}\alpha_{33})\beta_2 + (\alpha_{12}\alpha_{23} - \alpha_{13}\alpha_{22})\beta_3)/T \\ B &= ((\alpha_{13}\alpha_{23} - \alpha_{12}\alpha_{33})\beta_1 + (\alpha_{11}\alpha_{33} - \alpha_{13}^2)\beta_2 + (\alpha_{12}\alpha_{13} - \alpha_{11}\alpha_{23})\beta_3)/T \\ C &= ((\alpha_{12}\alpha_{23} - \alpha_{13}\alpha_{22})\beta_1 + (\alpha_{12}\alpha_{13} - \alpha_{11}\alpha_{23})\beta_2 + (\alpha_{11}\alpha_{22} - \alpha_{12}^2)\beta_3)/T \end{aligned}$$

If T is less than the threshold, the position angle is indeterminate. We set $B = 0$, and solve

$$\begin{aligned} T &\equiv \alpha_{11}\alpha_{33} - \alpha_{13}^2 \\ A &= (\alpha_{33}\beta_1 - \alpha_{13}\beta_3)/T \\ C &= (\alpha_{11}\beta_3 - \alpha_{13}\beta_1)/T \end{aligned}$$

If T is still less than the threshold, the beam fit is in serious trouble and the routine should probably issue an error message and return a default value. Alternatively, one could assume a circularly symmetric beam and use

$$A = C = \frac{\sum -(x^2 + y^2) \ln G}{\sum (x^2 + y^2)^2}$$

However determined, the quadratic coefficients are then converted back into elliptic beam parameters by calling program B.1 from Appendix B with $D = E = 0$ and $F = -\ln 2$. The accumulation region for the summations is somewhat arbitrary. Within *AIPS* and *SDE*, a pixel window of size 11×11 is used, centered on the peak. Only points greater than .35 of the peak are considered, and an attempt is made to consider only the points from the primary lobe of the beam, by starting at the central column in the window and working outwards. Highly elliptical beams with high sidelobes inclined near 45° could confuse these algorithms. If desired, a more robust algorithm could be devised for recognizing the primary lobe, but in such the cases the beam fit will still be problematical and the wise astronomer should examine it and override as needed. The physical meaning of the beam fit in such cases also becomes less clear, and questions of oversampling and the like are better answered in the u - v plane than in the sky plane.

The disadvantage of the linear algorithms is that the deviations of the true beam from the Gaussian model are compressed by the logarithm. For the purposes of calculating a residual flux scale where integrals over the beam are taken, the full nonlinear beam fitting is more appropriate in that it is less susceptible to systematically large deviations from the model. Nearly any nonlinear minimization routine will work, given a sufficiently accurate initial guess. Starting the nonlinear fit with the results of the linear fit works nicely. The *SDE* implementation of the beam fitting uses the Levenberg-Marquardt method, styled after the routine found in Press et al. (1986), and this has proved quite satisfactory. Typical differences in the fitted beam between the linear and nonlinear methods are only about 1% for well behaved data, but there is no reason not to do it correctly. Within *AIPS*, one can use the general purpose tasks *IMFIT* or *JMFIT* to perform a nonlinear fit. There is no provision for filtering high sidelobes in these tasks, but if it is problem they can be removed with *BLANK*.

A logic bug exists in the beam fitting algorithm in all versions of *AIPS* prior to 15JUL94. It results in either the major axis being underestimated or the minor axis overestimated, biasing all fits towards a more circular beam. In these versions, the nonlinear fitting is the only way to get an accurate answer.

Appendix D

Generalized Tapers and Beam Forcing

The elliptical Gaussian convolution calculations presented in Appendix B are more general than they might first appear. Convolving by a Gaussian in the image plane is multiplying by a Gaussian in the u - v plane is a taper. One might also wish to amplify the higher spatial frequencies and divide by a Gaussian instead of multiplying by one. It's not terribly productive to think of such an inverse taper as a convolution in the image plane, (the inverse Gaussian lacks a Fourier transform), but it's still a well posed problem in the u - v plane. Inverse tapering on visibility data is held finite by the limited support of any real sampling pattern.

The functional form of the Gaussian profile is $\exp(-Ax^2 + \dots)$, so by simply negating the sign of the quadratic parameters, the taper becomes an inverse taper. Examining the structure of equations B.4, the deconvolution of two elliptical Gaussians by solving for one of the inputs given the output is exactly the same as “convolution” of an elliptical Gaussian with an inverse taper and solving for the output. If the result of such a problem is defined in the image plane, it can be treated as ordinary deconvolution. But even if it is not defined in the image plane, the result of the deconvolution is still well defined in the u - v plane, and it may be applied to visibility data as a taper. This taper need not even be purely a downweighting or purely an upweighting. It is quite possible to have a taper along one direction in the u - v plane and an inverse taper along the orthogonal direction. In this case we have hyperbolic Gaussian tapering, since the curve of constant taper is a hyperbola rather than an ellipse. Though doubtless there are more general forms still, I call specification of the taper by signed quadratic coefficients, generalized tapering.

Any nondegenerate set of quadratic taper parameters can be reduced to a taper or inverse taper along two orthogonal axes and a rotation angle. **SDE** takes advantage of this fact in how it specifies the taper. A normal Gaussian taper is specified by the half widths and position angle of the equivalent convolution in the image plane. The only modification necessary to specify generalized tapers as well is the convention that the sign of the major and minor axes is transferred to the equivalent quadratic coefficient, as in

$$A_0 \equiv \text{sgn}(B_{maj})/B_{maj}^2 \quad C_0 \equiv \text{sgn}(B_{min})/B_{min}^2$$

It is fairly easy to use a generalized taper to force the fitted beam to any reasonable desired ellipse. The basic strategy is to approximate the initial and desired dirty

beam by their fitted Gaussians, and analytically deconvolve the dirty beam from the desired output. The resulting generalized taper is applied to the visibility data and the procedure iterated. Most well sampled cases will converge within 5–20 iterations of this simple recipe. Some cases will oscillate around the correct answer, and the cure here is to apply a convergence parameter to the taper correction determined by deconvolution. A simple linear scaling of the correction major & minor axes is sufficient for most cases. A more sophisticated convergence scheme could be used, one which keeps track of over and undershoots and automatically increases the damping when oscillation is detected, but this has not been investigated. In extremely poor coverage cases this algorithm might diverge, and for a given coverage there are values of the requested beam which no taper can produce. But with reasonable values of the requested beam this is usually a well behaved procedure.

The basic algorithm is explicitly given in Program D.1, and is implemented in **SDE** as the Perl script **findtaper**. The actual implementation is quite crude, being merely the tasks **uvmap** and **beamcalc** called in sequence. There are several minor problems associated with the implementation and one with the formalism. The latter problem is that quadratic taper parameters do not deal well with one-dimensional degenerate tapers — one or more coefficients become infinite. This is not a significant problem when specifying the taper directly, since one can always use an ellipse with an aspect ratio of 100. When called from **findtaper**, it is possible that a solution which converges exactly on one axis before the other might generate a one-dimensional correction, but this has never been observed with a real observation. **findtaper** merely assumes that if exact convergence is seen on either axis, then the complete solution has converged. The only concession made to the 1-D special case is liberal use of error trapping. The problems arising from the implementation as a shellscript rather than an integrated single task are a nontrivial source of roundoff error in the fixed format ASCII representation of the beam parameters used for communication between the tasks, many redundant calculations and much redundant IO between iterations, and an extremely crude termination criterion. However, even in spite of these problems, the procedure works quite well enough for practical use. An integrated beam forcing task would be preferable to the solution currently implemented in **SDE**, but any interferometric imaging packing should have provision for some sort of beam forcing. Clearly once generalized tapering is available within the mapping program, full beam forcing can be implemented with a minimum of programmer time.

```

begin                                     Find taper T which produces output beam O
    T := 0                                Initialize T or pass in best estimate from previous run
    factor := 1                             Set factor smaller if trouble converging
    B := evaluate_beam;
    for i := 1 to max_iter           Find  $B_{maj}, B_{min}, B_{pa}$  from initial taper
        C := deconvolve(B, O);
        if trap_error → done;          Finite main loop
         $C_{maj} := \text{factor } C_{maj}$ 
         $C_{min} := \text{factor } C_{min}$ 
        T := convolve(T, C);
        if trap_error → done;          Deconvolve B from O, giving a correction C
        L := B                           If error, assume that we're done
        B := evaluate_beam;
        if B = O → done;          Apply the correction giving the next taper estimate
        if B = L → done;          Save last value of Beam
    done :                               Find the new one
    end                                    Graceful exit
                                         No improvement in solution
                                         Reached iteration limit or found solution

```

Program D.1: Beam forcing iteration

Appendix E

Gridless Weighting Algorithm

The central step of a gridless weighting algorithm is to calculate the sum of the visibility weights in an elliptical neighborhood around each visibility. The simplest approach to implementing this would be a simple doubly nested loop. For each visibility we test every visibility in the data set and ask if it is close enough to the visibility under consideration to include in the local sum. The direct approach is far too slow for normal use, but it does demonstrate several implementation issues that must be addressed. The first is simply the test for “close enough”. In the usual algorithm for uniform weighting, the area under consideration is one cell in the gridded u - v plane. That is, a point u_i is within the neighborhood of u_k when $|u_i - u_{k0}| < \Delta u/2$, where $\Delta u = 1/2\ell_{max}$ and u_{k0} is the coordinate of the center of the grid cell containing u_k . A similar condition exists in v and m , so that the region of consideration is rectangular.

The gridless algorithm is more flexible, so we can use an elliptical region of consideration and also center the region properly around each visibility in turn. The test for proximity becomes

$$\frac{(u_i - u_k)^2}{r_u^2} + \frac{(v_i - v_k)^2}{r_v^2} \leq 1, \quad (\text{E.1})$$

where $r_u = \pi^{-1/2} \Delta u / \text{FOV}_w$ with Δu defined as above. The factor of $\pi^{-1/2}$ is so that the area of the consideration region is the same for both the gridded and gridless algorithms. FOV_w , the weighting field of view, is the fraction of the image size over which sidelobes are to be optimized. $\text{FOV}_w = 1$ is normal uniform weighting, while $\text{FOV}_w = 1/3$ would correspond to super-uniform weighting over a 3x3 u - v box. r_v is defined similarly.

The idea behind the gridless algorithm is to minimize the number of tests for proximity that must be made to determine all visibilities in the neighborhood of a given point. To this end we do an initial bucket sort of the visibilities. The result of this sort is an array of pointers to linked lists. Each entry of the array corresponds to a rectangular bin in the u - v plane, and all visibilities in that region are linked together in a singly linked list. This sort requires an arbitrary sized rectangular array of pointers, and an additional linear array of pointers of the same size as the visibility array. This binning operation is similar to gridding, with the important distinction that it is used only for efficiency. There are no approximations involved in the actual distance calculations between visibilities, and the final answer is independent of the binning parameters. From the basic parameters of the problem it is simple to calculate a small subset of the rectangular array containing all

possible visibilities in the neighborhood of (u_k, v_k) . We merely step over this subset region examining each head pointer in turn. If it is *nil*, there are no visibilities in that region. If it is *non-nil*, we step through the linked list examining each visibility for proximity to the target. Thus we must examine only the visibilities in the rough vicinity of the desired point. The finer we make the binning array, the more tightly we can approximate the desired neighborhood. We must also make more tests of the head pointer array, and the balance between these requirements determines the optimal size for a given problem. In Section E.2 we address the problem of selecting an appropriate bin size. A fixed bin size will be adequate for many purposes and is particularly simple to code.

A complication is introduced in that each visibility actually represents a Hermitian conjugate pair of visibilities, but it is only stored once in the visibility data base. For each visibility point $V_i(u_i, v_i)$, there is an implicit visibility point of $V_i^*(-u_i, -v_i)$ with the same weight. The algorithm must test for both signs of the conjugation, if there is any possibility of confusion. In some imaging packages the visibilities are gridded into a half plane, and the Hermitian nature of the data taken care of with a Real \Leftrightarrow Complex FFT routine. This is the most computationally efficient way to do the transform, but it introduces considerable complexity into the weighting algorithms. Our binning array need not bear any particular relationship to the grid used in the FFT of the data, and it is easiest to use an array that does not have a discontinuity at $u = 0$. The optimum size of the binning array can be quite large, possibly several thousand elements on a side — enough to be a potential problem for workstations already burdened with keeping the entire visibility set in virtual memory. We can reduce the memory requirements to nearly that of the half plane case by considering the positive ($u_i > 0$) conjugation of points to be weighted. In this case we need only consider neighboring points with $u_j > -r_u$, and do not need to bin any of the others. If we constrain the origin of the u - v plane to be between bins of the array, then the image of one bin through reflection in the origin exactly maps to another bin. Thus the same visibilities will be grouped together in both cases, and both head pointers can point to the same linked list. In addition, this choice guarantees that only a single conjugation of a given visibility can map to the same cell. Hence the algorithm must check only that it has the right conjugation for the given binning cell and include the weight or not as appropriate. Since in most cases the outer visibility track will cross $u = 0$, the minimum and maximum extent of v will be nearly equal. Thus it is no great penalty to impose that the v origin be between the center two bins. We will call the size of this array G_u by G_v , and map the u - v origin to between the bins u_{origin} & $u_{origin} + 1$ and similarly for v . G_v will be assumed even, with $v_{origin} = G_v/2$. The choice of u_{origin} is slightly more involved. The fundamental constraint is

$$\left\lceil \frac{|u_m| + \varepsilon}{\Delta G_u} \right\rceil + \left\lceil \frac{r_u + \varepsilon}{\Delta G_u} \right\rceil \leq G_u \quad (\text{E.2})$$

where ε is a small constant measured in cells which provides a guard region against roundoff error at the edge of the binning array. A conservative choice of ΔG_u and u_{origin} which risks

only wasting some of the binning array is

$$\Delta G_u = \frac{|u_m| + r_u}{G_u - 2} \quad u_{origin} = \left\lceil \frac{r_u + \varepsilon}{\Delta G_u} \right\rceil$$

The alternative is to simply select the smallest value of u_{origin} that satisfies equation E.2, trying each $\Delta G_u = (|u_m| + \varepsilon)/(G_u - u_{origin})$ in turn. The binning and weighting outlined in programs E.1 and E.2 will work with any reasonable value of u_{origin} .

```
begin
   $u_{scale} := \min((G_u - u_{origin} - \varepsilon)/u_{max}, (u_{origin} - \varepsilon)/r_u)$  Arrange scale so that  $u_{max}$  and  $v_{max}$  just fit on grid
   $v_{scale} := (v_{origin} - \varepsilon)/v_{max}$ 
  for  $k := 1$  to  $N_{vis}$ 
    if  $weight_k < 0 \rightarrow$  skip;
    if  $u_k \geq 0$  Ensure proper conjugation
       $u_{cell} := u_k * u_{scale}$ 
       $v_{cell} := v_k * v_{scale}$ 
    else
       $u_{cell} := -u_k * u_{scale}$ 
       $v_{cell} := -v_k * v_{scale}$ 
       $u_{bin} := \lfloor u_{cell} \rfloor + u_{origin} + 1$  Convert to cell index
       $v_{bin} := \lfloor v_{cell} \rfloor + v_{origin} + 1$  1 relative addressing used here
       $u'_{bin} := \lceil -u_{cell} \rceil + u_{origin} + 1$  Negative conjugation
       $v'_{bin} := \lceil -v_{cell} \rceil + v_{origin} + 1$ 
       $l := head_{u_{bin}, v_{bin}}$  Save the old head pointer
       $head_{u_{bin}, v_{bin}} := k$  Add the new visibility at the head
       $links_k := l$  Fix the link
      if  $u'_{bin} > 0$  Only some negative conjugations binned
         $head_{u'_{bin}, v'_{bin}} := k$  Negative conjugation points to the same list
    end
```

Program E.1: Visibility binning procedure.

Since weighting is a common operation, we are motivated to find the most efficient reasonable implementation possible. With another array of size N_{vis} and a bit more logic, the running time can be cut by another factor of several. The idea is to identify binning cells that are completely outside the region of consideration, so that we skip the cell without traversing the list. Additionally, the sum of all weights in each linked list are accumulated in the binning step, and when a cell is identified as being completely within the target neighborhood the summed weights are accumulated again without traversing the list. First, note that the bin-summed weights can be accumulated by simply adding the lines

```
if  $l = nil$ 
   $bin\_sum_k := weight_k$ 
else
   $bin\_sum_k := bin\_sum_l + weight_k$ 
```

at the end of the main loop in program E.1. After the initial binning, $head_{i,j}$ will point to the element of $weight_sum$ that contains the sum of all weights in the corresponding bin. Only the elements of $weight_sum$ that are pointed to by $head$ are used in the final weighting

```

begin
  for  $k := 1$  to  $N_{vis}$                                 Main loop of algorithm
    if  $weight_k < 0 \rightarrow skip;$ 
    if  $u_k \geq 0$                                      Select proper conjugation of target
       $u_{target} := u_k$ 
       $v_{target} := v_k$ 
    else
       $u_{target} := -u_k$ 
       $v_{target} := -v_k$ 
     $weight\_sum_k := 0$                                We find this for every visibility
     $bin\_low_u := \lfloor (u_{target} - r_u) * u_{scale} \rfloor + u_{origin} + 1$ 
     $bin\_high_u := \lfloor (u_{target} + r_u) * u_{scale} \rfloor + u_{origin} + 1$ 
     $bin\_low_v := \lceil (v_{target} - r_v) * v_{scale} \rceil + v_{origin} + 1$ 
     $bin\_high_v := \lceil (v_{target} + r_v) * v_{scale} \rceil + v_{origin} + 1$ 
    for  $i := bin\_low_u$  to  $bin\_high_u$                 Loop over the covering box
      for  $j := bin\_low_v$  to  $bin\_high_v$ 
         $l := head_{i,j}$                                 May be nil initially
        while  $l \neq nil$                                 Loop over any visibilities present
          if  $(u_l < 0 \wedge i \leq u_{origin}) \vee (u_l > 0 \wedge i > u_{origin})$ 
             $u_{test} := u_l$                             Ensure proper conjugation
             $v_{test} := v_l$ 
          else
             $u_{test} := -u_l$ 
             $v_{test} := -v_l$ 
            if inside $(u_{test}, v_{test})$            A fairly expensive test
               $weight\_sum_k := weight\_sum_k + weight_l$ 
               $l := links_l$                            Follow the link
  where
  funct inside $(u, v) \equiv \frac{(u - u_{target})^2}{r_u^2} + \frac{(v - v_{target})^2}{r_v^2} < 1.$        Centered around  $(u_{target}, v_{target})$ 
end

```

Program E.2: Simple gridless weighting procedure.

loop, but we have no way of knowing which elements will end up at the heads of the lists and the remainder of the array is convenient intermediate storage.

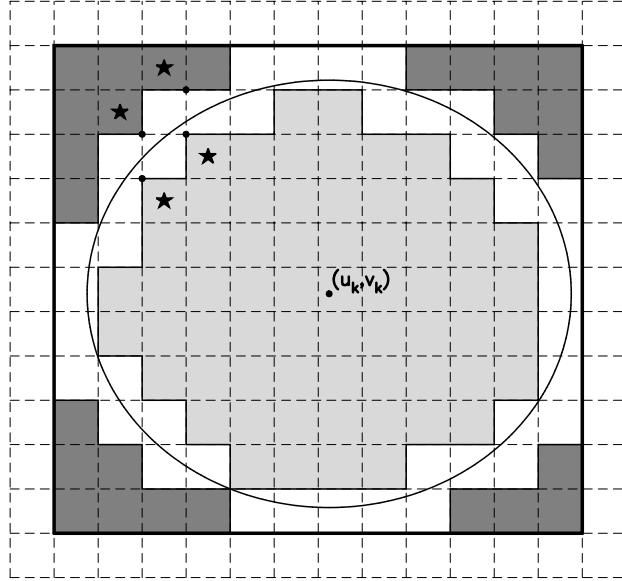


Figure E.1: Geometry of the binning array.

Consider Figure E.1 and program E.3. The heavy box in the figure represents the rectangular search area around (u_k, v_k) defined in program E.2. The lighter shaded bins are regions where all the weights within the bin should be included into the local density without needing to test each one individually. We simply test the corner of the bin known to be farthest from the origin of the ellipse. That is, in the upper left quadrant we test the upper left corner of the bin. If that is within the ellipse then the whole cell should be included. The darker shaded bins are those which should be skipped immediately. In a similar vein if the nearest corner of a given bin is outside the ellipse then the whole cell should be skipped. In the figure the starred bins have these corners marked. The remaining bins are the only ones where we must search the list in the style of program E.2. These latter individual comparisons are responsible for the $O(N^2)$ behavior of the algorithm, and do indeed constitute the majority of the run time for typical sized problems. In program E.3 we present the central decision tree required to implement this. For problems where the u - v coordinates, visibility weights and temporary arrays can fit into virtual memory, this algorithm is entirely satisfactory for calculation of gridless local weight density.

E.1 Variations of the Algorithm

In program E.3 every binning cell is examined individually for occupancy, containment and exclusion from the target ellipse. An alternative approach is to calculate the intersection of the lines separating the cells in v with the ellipse. By working upwards and

```

begin
  if headi,j = nil → skip;
  u := (i - uorigin) * uscale
  v := (j - vorigin + 1) * vscale
  if u + Δu < utarget then
    if v - Δv > vtarget then
      if inside(u, v) → sum_weights;
      elseif outside(u + Δu, v - Δv) → skip;
      else search_list;
    elseif v < vtarget then
      if inside(u, v - Δv) → sum_weights;
      elseif outside(u + Δu, v) → skip;
      else search_list;
    else
      if inside(u, v) ∧ inside(u, v - Δv) → sum_weights;
      else search_list;
  elseif u > utarget then
    if v - Δv > vtarget then
      if inside(u + Δu, v) → sum_weights;
      elseif outside(u, v - Δv) → skip;
      else search_list;
    elseif v < vtarget then
      if inside(u + Δu, v - Δv) → sum_weights;
      elseif outside(u, v) → skip;
      else search_list;
    else
      if inside(u + Δu, v) → sum_weights;
      elseif outside(u + Δu, v - Δv) → skip;
      else search_list;
  else
    if v - Δv > vtarget then
      if inside(u, v) ∧ inside(u + Δu, v) → sum_weights;
    elseif v < vtarget then
      if inside(u, v - Δv) ∧ inside(u + Δu, v - Δv) → sum_weights;
    else
      if inside(u, v) ∧ inside(u + Δu, v) ∧
          inside(u + Δu, v - Δv) ∧ inside(u, v - Δv) → sum_weights;
      else search_list;
  where
  funct outside(u, v) ≡ ¬inside(u, v).
  funct sum_weights ≡ weight_sumk := weight_sumk + bin_sumk; skip; .
  funct search_list ≡ (as in inner loop of program D.2).
end

```

Same loop structure as program D.2
Most bins take this case
Fiducial point of bin is upper right corner

Left hand side of ellipse
Upper left quadrant

Lower left quadrant

Straddles center in v

Right hand side of ellipse
Upper right quadrant

Lower right quadrant

Straddles center in v

Straddles center in u
Upper half of ellipse

Lower half of ellipse

Cell contains center

Program E.3: Central decision tree for quick gridless weighting

downwards from the center lines, one can calculate exactly the limits of the interior and border cells, which would appear to be much more efficient than testing each cell. In practice, however, the filling factor of the binning array is between 5 and 20 percent. Testing a cell for occupancy is very quick, and must be done in any event even in the calculated limit variation. In the region of the optimum running time, the ellipse size is near 15 points in diameter, so a comparable number of elliptical calculations are done in each approach. The direct testing approach is more straightforward to code and slightly faster than calculated limits, so there is no need to consider this variation further.

The original version developed used strict half plane binning. That is, no conjugations with $u_i < 0$ were binned at all. All negative conjugations were found by searching the corresponding region of the positive half plane. The problem is that a rectangular region including the origin decomposes into four distinct rectangular subregions in the positive conjugation half plane, with the visibility conjugation in the center regions indeterminate. This version used slightly less memory for the binning array, but was somewhat slower due to the redundant checks for conjugation in the central region and much more complicated to code. The subroutine containing the inner loop was over 1000 lines of FORTRAN long! This approach is not recommended.

If performance is paramount and memory cheap, one could consider separating the conjugations. Each visibility would be entered in the database twice, once for each conjugation, and there would be no need to decode the conjugations in the inner loop. It is unlikely that this would increase the speed by more than perhaps 10%, however. The binning array would need to be full sized, and the size of the visibility and link arrays doubled or tripled. For contemporary machines this does not seem a useful tradeoff.

The recommended algorithm assumes that all relevant arrays can fit into virtual memory. For all but the largest problems, this is a reasonable assumption. Still, for some purposes it may be desirable to create a disk based version that will handle problems of larger or arbitrary sizes. Assume that the maximum number of visibilities that fall into a strip of width $w \equiv \lceil 2r_u/\Delta u \rceil + 1$ bins wide is N'_{vis} . If it is practical to hold in memory visibility arrays of size $N'_{vis} \ll N_{vis}$ and also a head pointer array of size w by G_v , one can proceed quite straightforwardly. The visibilities are first sorted into ascending u . The visibilities are binned into the array, mapping the strip $-r_u + (k - 1)\Delta u < u \leq -r_u + k\Delta u$ into the bin $(k \bmod w) + 1$, stopping at the first visibility where $u_k > -r_u + w\Delta u$. The local weight densities for the visibilities within the strip $0 < u < \Delta u$ can now be calculated as before. The *link* and coordinate arrays are scanned, and any link into the leftmost active strip is replaced by *nil*, which trims these visibilities from the binned storage. The $(k+1)$ th portion of the u - v data file corresponding to a single strip of width Δu is read in and binned to column $(k + 1 \bmod w)$, replacing the previous contents which are no longer needed. The next strip of densities is calculated and the process repeated. The links are still the visibility indices, but all references to the *link* and coordinate arrays are now taken modulo N'_{vis} .

If memory is insufficient for even these reduced buffers, the visibilities could be sorted in v as well as u , arranged so that all visibilities within a bin are contiguous. A similar strategy as above would be employed to work along the v axis for each set of u . The house keeping would be involved to say the least, and it is difficult to imagine a scenario where this would be required.

There is an advantage to double sorting even for the primary algorithm that we have advocated above. If many visibility tracks cross the same binning cell, the resulting linked list of visibilities will stripe across memory with a wide stride, possibly resulting in excessive page faulting. The best possible sort to cure this problem would be the one described above which knows about the geometry of the binning array. But even a general purpose sort such as UVSRT from *AIPS* should serve the purpose of roughly localizing the visibilities in memory. This assertion has not been tested, however, as page faulting has not been found to be a problem even with moderately large data sets processed on a Sparc IPX with 32 MB of memory. A 336,000 visibility data set showed no page faulting on the performance meter and had identical run times for Time-Baseline and u - v sorted data sets.

E.2 Automatic Bin Size Selection

A consideration that has been deferred in the above discussion is exactly how large the binning array should be. If the array is too small, then the algorithm will spend inordinate amounts of time searching the linked lists on the edge of the ellipse, and will not be able to exploit the cached nature of the summed weights interior to the ellipse. If the array is too large, then the algorithm will spend too much time checking array cells for containment and exclusion, though fractionally fewer cells will fall on the ellipse boundary. With an analytic model of the algorithm run time, one could select an appropriate bin size by simply minimizing the expected run time.

A careful analytic model was developed of the algorithm's run time, based on assignment of arbitrary costs to each branch that the inner and outer loops can take, numerical calculation of the probabilities of each branch from the parameters of the problem, elimination of linear dependencies in the model, and then least squares determination of the resulting combined constants by fitting the model to a variety of measured run times. The hope was that such a model would perform well when extrapolated to regions of parameter space untested. However, even with a singular value analysis of the fitting matrix, the resulting constants were not terribly stable against determination from different subsets of the data. It required a somewhat arbitrary selection of cutoff in singular value to obtain constants that appeared physically reasonable. As the minimum of expected run time against array size is fairly wide, all of the various analytic models performed well enough for practical purposes. The algorithms performed well enough that the calculation of the analytic minimum was consuming more time than saved by the use of the true optimum grid size compared to relatively crude estimates. The simple optimum grid size estimates

themselves were surprisingly good and comparable to the estimates from the full model, so there is no need for the details of the analytic model to be presented here.

Examining the optimum solutions found by direct exploration of the parameter space, it was noted that the ratio $k_G \equiv \sqrt{(r_u/\Delta G_u)(r_v/\Delta G_v)} \equiv \sqrt{k_{Gu}k_{Gv}}$ was remarkably constant. While it is intuitively clear that the radius of the weighting ellipse in binning cells should be important, it is not as clear that this quantity should completely dominate the run time performance. None the less, k_G ranged from 6.3 to 8.9 over a wide range of imaging parameters, for both data sets, on both a Sun IPX and an IBM RS/6000! If one simply assumes that $k_{Gu} = k_{Gv} = 7.6$ is an approximate constant of the algorithm, one can easily derive a rule for optimum grid size in terms of imaging parameters.

$$\frac{r_v}{\Delta G_v} = \frac{r_v}{v_{max}/(\frac{G_v}{2} - \varepsilon)} = k_G$$

$$G_v \approx 2k_G \sqrt{\pi} v_{max} FOV_w / \Delta v \approx 27 v_{max} FOV_w / \Delta v \quad (\text{E.3})$$

All of these quantities are easily available to a program doing the reweighting. For the purposes of estimation, consider that the uniformly weighted FWHM beamwidth of a typical full synthesis observation is η/v_{max} , where η ranges from about .75 for the ATCA to .95 for the VLA. Remembering that the full angular width of the image is $1/\Delta v$, we define the image field of view in beamwidths as $FOV_i \equiv v_{max}/\eta\Delta v$.

$$G_v \approx 2k_G \eta \sqrt{\pi} FOV_i FOV_w \approx 23 FOV_i FOV_w$$

The optimum bin size goes as the image field of view measured in uniform beamwidths times the weighting field of view.

G_u is determined similarly, and yields

$$G_u \approx k_G \sqrt{\pi} u_{max} FOV_w / \Delta u + [k_G] \approx 11.5 FOV_i FOV_w + 8 \quad (\text{E.4})$$

Needless to say, a reasonable value of k_G should be determined experimentally for any new implementation of this algorithm, but it seems likely that these values will not be too far off. And of course, the worst that will happen if they are is that the weighting algorithm will run slower than optimum.

The primary data sets used for run time determinations were a 336,000 visibility set from the A configuration VLA, and a 110,000 visibility set from the Australia Telescope Compact Array. Both were from high declination observations and the $u-v$ coverage fairly circular. The VLA data was sampled at roughly 3.5 pixels per uniform beam, (512^2 image, $\Delta u = \Delta v = .05''$, $v_{max}/\hat{v}_{max} = .46$). The AT data was highly oversampled at roughly 9.5 pixels per beam, (512^2 image, $\Delta u = \Delta v = .1''$, $v_{max}/\hat{v}_{max} = .17$). The oversampling means that the run times for the AT case are comparable to that of a more conservative sampling and a FOV_w twice as large. Some example run times as a function of binning array size are given in Table E.2. For these runs, $G_u = [G_v * u_{max}/v_{max} + 8.5]$, which is essentially

equivalent to equations E.3 and E.4 near the run time minima. The machines used were a Sun Sparc IPX with 32 MB of memory, and an IBM RS/6000-560 with heaps of it.

The most germane question is how well the algorithm performs with the binning array size estimates as compared to the optimum size found by direct examination of the parameter space. For each case in Table E.2, a more densely sampled version of these data were fitted with a parabola in the region of the minimum to determine the optimum binning array size. The full analytic model (with coefficients determined from the composite data set, but separately for each processor) and the $k_G = 7.6$ simple expression were then used to predict the optimum binning array size. The run time excesses tabulated are simply the fitted parabolae evaluated at the estimated array sizes. The points to note from these tables are:

- *Run Times for typical parameters and data sets are slower than gridded weighting, but fast enough for routine use*
- *The simple $k_G = \text{const}$ method of estimating an optimum binning array size is entirely adequate*

Less important points are

- There is a strong dependence of run time on weighting field of view, and a weaker dependence on the size of the binning array
- Both the simple estimation and the full analytic model predict the optimum binning array size well enough that the run times are within a few percent of optimum.
- The simple method of estimation makes its maximum error in regions of the highest run time, while the full model may make a significant error anywhere

Clearly there is no great advantage to the sophisticated modelling. It might eventually prove useful as a diagnostic in cases where the simple approach fails, but until such a case arises, its use is not recommended.

System	Dataset	FOV_w	Gridsize (G_v)				
			250	500	1000	1500	2000
IPX	AT	1.00	279.4	172.5	135.7	152.9	190.2
IPX	AT	0.50	391.7	245.1	259.9	375.1	545.4
IPX	AT	0.25	985.7	682.9	827.1	1280.3	1925.0
IPX	VLA	1.00	1727.1	934.9	582.7	508.9	511.7
IPX	VLA	0.50	3197.1	1753.9	1197.1	1173.4	1309.4
IPX	VLA	0.25	6559.6	3818.2	2976.7	3303.1	4026.1
IBM	AT	1.00	59.1	38.1	32.2	36.7	45.3
IBM	AT	0.50	84.7	56.4	62.5	87.0	121.3
IBM	AT	0.25	211.7	157.7	195.0	289.1	418.3
IBM	VLA	1.00	510.0	275.5	172.9	152.2	152.4
IBM	VLA	0.50	941.2	520.7	361.9	355.4	387.5
IBM	VLA	0.25	1930.4	1146.1	917.1	1005.8	1185.5

Table E.1: Running times for the gridless weighting algorithm main loop. Times are in seconds.

System	Dataset	FOV_w	k_{Gu}	k_{Gv}	Gv_{min}	RT_{min}	Gv_{mod}	ΔRT_{mod}	Gv_{simp}	ΔRT_{simp}
IPX	AT	1.00	6.86	7.41	1075.92	134.61	1228.52	2.55	1147.55	0.56
IPX	AT	0.50	7.31	7.89	810.71	224.21	989.85	15.78	811.73	0.00
IPX	AT	0.25	8.65	9.34	678.34	658.74	819.01	34.15	574.28	18.69
IPX	VLA	1.00	6.40	6.86	1731.80	503.32	1955.65	6.07	1987.27	7.90
IPX	VLA	0.50	7.03	7.53	1344.30	1148.93	1414.54	2.67	1405.50	2.03
IPX	VLA	0.25	8.06	8.62	1089.58	2936.66	1080.84	0.18	994.13	21.91
IBM	AT	1.00	6.49	7.01	1017.26	32.05	1434.35	3.69	1147.55	0.36
IBM	AT	0.50	7.07	7.63	783.72	54.00	1134.87	12.16	811.73	0.08
IBM	AT	0.25	8.29	8.95	650.27	156.03	884.02	18.30	574.28	1.93
IBM	VLA	1.00	6.48	6.93	1751.38	150.61	2252.87	7.67	1987.27	1.70
IBM	VLA	0.50	7.09	7.59	1356.19	348.39	1457.12	1.47	1405.50	0.35
IBM	VLA	0.25	8.05	8.62	1088.69	906.74	1077.58	0.08	994.13	5.84

Table E.2: Run time excesses using optimum binning array size approximations. Run times are in seconds, array sizes in cells, and the rest are dimensionless.

Appendix F

Complex Fourier Series

Complex Fourier Series are used less frequently than their real counterparts, but provide elegant solutions to some problems.

The basic orthogonality of the basis function is

$$\int_0^{2\pi} (e^{im\theta})^* e^{in\theta} d\theta = 2\pi \delta_{mn}$$

A complex valued function $f(\theta)$ on $(-\pi, \pi)$ can be mapped to the unit circle by the function $z = e^{i\theta}$. Let

$$f(\theta) = g(z) = g(e^{i\theta}) = \sum_{n=-\infty}^{\infty} d_n e^{in\theta}$$

$$\int_{-\pi}^{\pi} f(\theta) e^{-im\theta} d\theta = \sum_{n=-\infty}^{\infty} d_n \int_{-\pi}^{\pi} e^{-im\theta} e^{in\theta} d\theta = 2\pi \sum_{n=-\infty}^{\infty} d_n \delta_{mn} = 2\pi d_m$$

After changing variables to $\theta/\pi = u/u_m$ and noting that we normally work with $V(u) \equiv f(\pi u/u_m)$, we have derived the expansion

$$V(u) = \sum_{n=-\infty}^{\infty} a_n e^{in\theta}$$

where $a_n = \frac{1}{2u_m} \int_{-u_m}^{u_m} e^{-in\pi u/u_m} V(u) du$

Appendix G

SDE

The Software Development Environment (**SDE**) is an attempt to provide a simple, friendly environment for the *development* of imaging algorithms. It has many strange and wonderful algorithms not found in other packages. The useful ones are supposed to migrate to more common packages, but it doesn't always work that way.

SDE is available for ftp, but *not supported* by the authors or by NRAO. Comments and questions may be sent by email to `dbriggs@nrao.edu` or `tcornwel@nrao.edu`, but no guarantees are made as to subsequent action.

SDE is characterized by relatively close ties to the unix family of operating systems and by the assumption that virtual memory is plentiful. Rather than reinvent the operating system and insulate the programmer from the vagaries of many divergent platforms, **SDE** sacrifices some portability in the name of simplicity and relies as much as possible on native tools and OS support. The assumption of plentiful physical and virtual memory leads to the common practice of loading all image and visibility data completely into memory for the duration of their use. This greatly reduces the burden of memory management on the programmer and has the side effect of allowing algorithms to proceed at maximum speed in the common case where all data can fit within physical memory. The tradeoff is that there will be some problems that will not fit into virtual memory which might be handled by a sequential disk based philosophy. Fortunately this is rarely a problem.

The current **SDE** distribution is available by anonymous ftp from the host `ftp.aoc.nrao.edu` (146.88.1.103), in the `pub/sde` subdirectory. This distribution is merely a snapshot of the **SDE** source directory tree, along with various tools and installation instructions. It is not a “polished” release.

Many **SDE** routines make use of the Numerical Recipes (Press et al., 1986) library, which for copyright reasons we cannot distribute. The library is available for modest cost from Cambridge University Press. Ordering information may be found in any of the Numerical Recipes family of books.

In addition, the **SDE** programs which produce graphical output use the **PGPLOT** graphics library, available from the Caltech astronomy department. Currently the package may be obtained by anonymous ftp from the host `deimos.caltech.edu` (131.215.139.14), in the **PGPLOT** subdirectory.

Appendix H

Electronic Supplement

It is not practical to present the simulations from Chapter 5 over their entire parameter space. In addition, there are color versions of several figures which are interesting, but not sufficiently so to sacrifice the black & white nature of this dissertation. This additional material has been collected into the electronic Appendices section of the dissertation, and can be obtained as part of the **SDE** distribution described in Appendix G.

The Appendices can be obtained from the directory `data/dbriggs`, or as part of the **SDE** `.tar` files. This distribution might change slightly with time, so check the file `README` in this directory for the current contents.

Bibliography

- Anantharamiah, K. R., Cornwell, T. J., and Narayan, R. (1989). Synthesis imaging of spatially coherent objects. In (Perley et al., 1989), chapter 22, pages 415–430.
- Anderson, E. et al. (1992). *LAPACK Users Guide*. Society for Industrial and Applied Mathematics, Philadelphia.
- Andrews, H. C. and Hunt, B. R. (1977). *Digital Image Restoration*. Signal Processing. Prentice Hall, Englewood Cliffs, New Jersey.
- Apostol, T. M. (1969). *Calculus*, volume II. John Wiley & Sons, New York.
- Arfken, G. (1970). *Mathematical Methods for Physicists*. Academic Press, New York, 2nd edition.
- Bracewell, R. N. (1986). *The Fourier Transform and Its Application*. McGraw Hill, New York, 2nd edition.
- Briggs, D. S. (1994). Superresolution of sn 1987a: A comparative deconvolution study. In Hanisch, R. J. and White, R. L., editors, *The Restoration of HST Image and Spectra – II*, pages 250–256, 3700 San Martin Drive, Baltimore, MD 21218. Space Telescope Science Institute.
- Briggs, D. S. and Cornwell, T. J. (1994). Clean/mem deconvolution errors: semicompact sources. In Robertson, J. G. and Tango, W. J., editors, *Very High Angular Resolution Imaging*, number 158 in International Astronomical Union Symposium, pages 212–214, Dordrecht, The Netherlands. Kluwer Academic Publishers.
- Briggs, D. S., Davis, R. J., Conway, J. E., and Walker, R. C. (1994). High dynamic range imaging with the vla. VLBA Memo 671, National Radio Astronomy Observatory, Socorro, New Mexico.
- Christiansen, W. N. and Högbom, J. A. (1985). *Radiotelescopes*. Cambridge University Press, Cambridge, 2nd edition.
- Clark, B. G. (1980). An efficient implementation of the algorithm “clean”. *Astronomy and Astrophysics*, 89:377–378.
- Clark, B. G. (1989). Coherence in radio astronomy. In (Perley et al., 1989), chapter 1, pages 1–9.
- Cohen, J. G. (1991). Tests of the photometric accuracy of image restoration using the maximum entropy algorithm. *Astronomical Journal*, 101(2):734–737.

- Conley, R. W., Payne, D. M., Restaino, S. R., and Loos, G. C. (1994). Synthetic images from simulations of long baseline 2d optical interferometers. Technical report, U.S. Air Force Phillips Laboratory, PL/LIMI 3550 Aberdeen SE, KAFB, NM 87117.
- Cornwell, T. and Braun, R. (1989). Deconvolution. In (Perley et al., 1989), pages 167–183.
- Cornwell, T. J. (1981). An error analysis of calibration. VLA Scientific Memo 135, National Radio Astronomy Observatory, Socorro, New Mexico.
- Cornwell, T. J. (1983). A method of stabilizing the clean algorithm. *Astronomy and Astrophysics*, 121:281–285.
- Cornwell, T. J. and Evans, K. J. (1985). A simple maximum entropy deconvolution algorithm. *Astronomy and Astrophysics*, 143:77–83.
- Cornwell, T. J. (1984). Brightness temperature limits for filled and unfilled apertures. Millimeter Array Memo 24, National Radio Astronomy Observatory, Socorro, New Mexico.
- Crane, P. C. and Napier, P. J. (1989). Sensitivity. In (Perley et al., 1989), chapter 7, pages 139–165.
- Dwarakanath, K. S., Deshpande, A. A., and Shankar, N. U. (1990). A modified algorithm for cleaning wide-field maps with extended structures. *Journal of Astrophysics and Astronomy*, 11:311–322.
- Efron, B. and Tibshirani, R. (1991). Statistical data analysis in the computer age. *Science*, 253:390–395.
- Fanaroff, B. L. and Riley, J. M. (1974). The morphology of extragalactic radio sources of high and low luminosity. *Monthly Notices of the Royal Astronomical Society*, 167:31P–35P.
- Felli, M. and Spencer, R. E., editors (1989). *Very Long Baseline Interferometry, Techniques and Applications*, volume 283 of *Series C: Mathematical and Physical Sciences*. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Frieden, B. R. (1972). Restoring with maximum likelihood and maximum entropy. *Journal of the Optical Society of America*, 62(4):511–518.
- Gerchberg, R. W. and Saxton, W. O. (1972). A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237–246.
- Golub, G. and Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *SIAM Journal of Numerical Analysis (Series B)*, 2(2):205–224.

- Gonzalez, R. C. and Woods, R. E. (1992). *Digital Image Processing*. Addison-Wesley Publishing Company.
- Gull, S. F. and Daniell, G. J. (1978). Image reconstruction from incomplete and noisy data. *Nature*, 272:686–690.
- Högbom, J. A. (1974). Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics Supplement*, 15:417–426.
- Hunt, B. R. (1971). A matrix theory proof of the discrete convolution theorem. *IEEE Transactions on Audio and Electroacoustics*, 19:285–288.
- Hunt, B. R. (1973). The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, C-22(9):805–812.
- Jackson, J. D. (1975). *Classical Electrodynamics*. John Wiley & Sons, Inc., New York, 2nd edition.
- Jakobsen, P., Albrecht, R., Barbieri, C., et al. (1991). First results from the faint object camera: Sn 1987a. *Astrophysical Journal (Letters)*, 369:L63–L66.
- Jaynes, E. T. (1982). On the rationale of maximum-entropy methods. *Proceedings of the IEEE*, 70(9):939–952.
- Keel, W. C. (1988). The optical continua of extragalactic radio jets. *Astrophysical Journal*, 329:532–550.
- Kendall, M. K. and Stuart, A. (1977). *The Advanced Theory of Statistics*. MacMillan Publishing Company, New York, 4th edition.
- Lapidus, L. and Pinder, G. F. (1982). *Numerical Solution of Partial Differential Equations in Science and Engineering*. John Wiley and Sons, New York.
- Lawson, C. L. and Hanson, R. J. (1974). *Solving Least Squares Problems*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- Lucy, L. B. (1974). An iterative technique for the rectification of observed distributions. *Astronomical Journal*, 79(6):745–754.
- Marsh, K. A. and Richardson, J. M. (1987). The objective function implicit in the clean algorithm. *Astronomy and Astrophysics*, 182:174–178.
- Narayan, R. and Nityananda, R. (1984). Maximum entropy – flexibility versus fundamentalism. In (Roberts, 1984), pages 281–290.
- Narayan, R. and Nityananda, R. (1986). Maximum entropy image restoration in astronomy. *Annual Reviews of Astronomy and Astrophysics*, 24:127–70.

National Radio Astronomy Observatory (1990). The green bank telescope (a telescope for the twenty-first century: Request for proposals. Green Bank, WV.

Noble, B. and Daniel, J. W. (1977). *Applied Linear Algebra*. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Papoulis, A. (1975). A new algorithm in spectral analysis and band-limited extrapolation. *IEEE Transactions on Circuits and Systems*, CAS-22(9):735–742.

Perley, R. A. (1989). High dynamic range imaging. In (Perley et al., 1989), chapter 16, pages 287–311.

Perley, R. A., Schwab, F. R., and Bridle, A. H., editors (1989). *Synthesis Imaging in Radio Astronomy*, volume 6 of *Astronomy Society of the Pacific Conference Series*. Astronomical Society of the Pacific, 390 Ashton Ave, San Francisco, CA 94112.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1986). *Numerical Recipes*. Cambridge University Press, 1st edition.

Puetter, R. C. and Piña, R. K. (1994). Beyond maximum entropy: Fractal pixon-based image reconstruction. *Experimental Astronomy*, 3:293–296.

Raisbeck, G. (1955). The order of magnitude of the fourier coefficients in functions having isolated singularities. *American Mathematical Monthly*, 62:149–155.

Richardson, W. H. (1972). Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America*, 62(1):55–59.

Roberts, J. A., editor (1984). *Indirect Imaging*. Cambridge University Press.

Rushforth, C. K. (1987). Signal restoration and fredholm integral equations. In Stark, H., editor, *Image Recovery: Theory and Application*, chapter 1, pages 1–27. Academic Press, Inc., Orlando, FL.

Sault, R. J. (1984). The weighting of visibility data. VLA Scientific Memorandum 154, NRAO.

Schwab, F. R. (1984). Relaxing the isoplanaticism assumption in self-calibration; applications to low-frequency radio interferometry. *Astronomical Journal*, 89:1076–1081.

Schwarz, U. J. (1978). Mathematical-statistical description of the iterative beam removing technique (method clean). *Astronomy and Astrophysics*, 65:345–356.

Schwarz, U. J. (1979). The method clean – use, misuse, and variations. In van Schooneveld, C., editor, *Image Formation from Coherence Functions in Astronomy*, number 49 in International Astronomical Union Colloquium, pages 261–275, Dordrecht, The Netherlands. D. Reidel Publishing Company.

- Schwarz, U. J. (1984). The reliability of clean maps and the corrugation effect. In (Roberts, 1984), pages 255–260.
- Skilling, J. and Bryan, R. K. (1984). Maximum entropy image reconstruction: general algorithm. *Monthly Notices of the Royal Astronomical Society*, 21:111–124.
- Sramek, R. A. and Schwab, F. R. (1989). *Imaging*, chapter 6, pages 117–138. Volume 6 of (Perley et al., 1989).
- Staveley-Smith, L., Briggs, D. S., Rowe, A. C. H., et al. (1993a). Structure of the radio remnant of supernova 1987a. *Nature*, 366:136–138.
- Staveley-Smith, L., Manchester, R. N., Kesteven, M. J., et al. (1993b). Structure in the radio remnant of supernova 1987a. *Proceedings of the Astronomical Society of Australia*, 10(4):331–334.
- Staveley-Smith, L., Manchester, R. N., Tzioumis, A. K., et al. (1993c). Radio emission from sn 1987a. In *IAU Colloquium 145*, number 145 in IAU Colloquium, page in press, Xian, China.
- Steer, D. G., Dewdney, P. E., and Ito, M. R. (1984). Enhancements to the deconvolution algorithm “clean”. *Astronomy and Astrophysics*, 137:159–165.
- Swarztrauber, P. N. (1985). *fftpack, a package of fortran subprograms for the fast fourier-transform of periodic and other symmetric sequences*. National Center for Atmospheric Research, Boulder, CO, version 4.
- Tan, S. M. (1986). An analysis of the properties of clean and smoothness stabilized clean – some warnings. *Monthly Notices of the Royal Astronomical Society*, 220:971–1001.
- Thompson, A. R. (1989). The interferometer in practice. In (Perley et al., 1989), chapter 2, pages 11–34.
- Thompson, A. R., Clark, B. G., Wade, C. M., and Napier, P. J. (1980). The very large array. *Astrophysical Journal Supplement Series*, 44:151–167.
- Thompson, A. R., Moran, J. M., and Swenson, Jr., G. W. (1986). *Interferometry and Synthesis in Radio Astronomy*. John Wiley & Sons, New York.
- Wakker, B. P. and Schwarz, U. J. (1988). The multi-resolution clean and its application to the short-spacing problem in interferometry. *Astronomy and Astrophysics*, 200:312–322.
- Walker, R. C. (1989). Sensitivity. In (Felli and Spencer, 1989), chapter 9, pages 163–182.
- Wernecke, S. J. and D’Addario, L. R. (1976). Maximum entropy image reconstruction. *IEEE Transactions on Computers*, C-26:351–364.

- White, R. L. (1993). Improvements to the richardson-lucy method. In *Image Restoration Newsletter*, pages 11–23. Space Telescope Science Institute.
- Wiener, N. (1942). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. MIT Press, Cambridge, Massachusetts.
- Wieringa, M. (1991). *327 MHz Studies of the High Redshift Universe and the Galactic Foreground*. PhD thesis, University of Leiden.
- Wilczek, R. and Drapatz, S. (1985). A high accuracy algorithm for maximum entropy image restoration in the case of small data sets. *Astronomy and Astrophysics*, pages 9–12.
- Wild, J. P. (1970). De-convolution of barely resolved radio sources mapped with aerial beams of elliptical cross section. *Australian Journal of Physics*, 23:113–115.
- Wilkinson, P. N. (1989). An introduction to closure phase and self-calibration. In (Felli and Spencer, 1989), pages 69–93.
- Wilkinson, P. N., Tzioumis, A. K., Akujor, C. E., et al. (1990). High-quality images of the css quasars 3c48 and 3c380. In Zensus, J. A. and Pearson, T. J., editors, *Parsec Scale Radio Jets*, pages 152–160. Cambridge University Press.
- Wilkinson, P. N., Tzioumis, A. K., Benson, J. M., et al. (1991). A disrupted radio jet inside the host galaxy of the quasar 3c48. *Nature*, 352:313–315.

*High Flight**by John G. Magee*

*Oh, I have slipped the surly bonds of earth
And danced the skies on laughter-silvered wings;
Sunward I've climbed, and joined the tumbling mirth
Of sun-split clouds – and done a hundred things
You have not dreamed of – wheeled and soared and swung
High in the sunlit silence. Hovering there.
I've chased the shouting wind along, and flung
My eager craft through footless halls of air.
Up, up the long, delirious, burning blue
I've topped the windswept heights with easy grace
Where never lark, or even eagle flew.
And, while with silent, lifting mind I've trod
The high untrespassed sanctity of space,
Put out my hand, and touched the face of God.*