Geostatistical Techniques for Practical Wireless Network Coverage Mapping

by

Caleb Timothy Phillips

A.S., Portland Community College, 2002

B.S., Portland State University, 2006

M.S., University of Colorado, 2010

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

2012

This thesis entitled:
Geostatistical Techniques for
Practical Wireless Network Coverage Mapping
written by Caleb Timothy Phillips
has been approved for the Department of Computer Science

_____

Prof. Douglas Sicker

_____

Prof. Dirk Grunwald

Date _____

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Phillips, Caleb Timothy (Ph.D., Computer Science)

Geostatistical Techniques for

Practical Wireless Network Coverage Mapping

Thesis directed by Prof. Douglas Sicker

The problem of mapping the extent of "usable" coverage of an existing wireless network is important in a large number of applications, including communicating the abilities of the network to users, identifying coverage gaps and planning expansion, discovering opportunities for spectrum reuse, and determining possible sources of interference with other networks. This thesis addresses fundamental but unsolved problems of measurement-based wireless coverage mapping: where should measurements be made, how many are necessary, and what can be said about the coverage at points that have not been measured. To address these problems, this thesis advocates a geostatistical approach using optimized spatial sampling and ordinary Kriging. A complete system for coverage mapping is developed that systematically adresses measurement, sampling, spatial modeling, interpolation, and visualization. This geostatistical method is able to produce more accurate and robust coverage maps than the current state of the art methods, and is able to discover coverage holes as effectively as dedicated heuristic methods using a small number of measurements. Several important practical extensions are investigated: applying these methods to drive-test measurements which have been resampled to alleviate effects from sampling bias, and crowd-sourced coverage mapping applications where volunteer-collected measurements may be sparse or infrequent. The resulting maps can then be refined iteratively, and updated systematically over time using an optimized iterative sampling scheme. An extensive validation is performed using measurements of production WiFi, WiMax, GSM, and LTE networks in representative urban and suburban outdoor environments.

Dedication

To my grandmother, Lila May Hiatt (Richards)

Acknowledgements

Thanks first, to the denizens of the systems laboratory and most especially my “elder brothers” in research, Eric Anderson and Gary Yee. Thanks to Prof. Suresh Singh who provided me the first opportunity to do interesting research and walked with me both up mountains and hard research problems. Thanks to Russell Senior who taught me about patience and care in research and data collection and inspired me with his tireless rigor and curiosity. Thanks to my fellow volunteers at the Personal Telco Project who inspired me with their passion for wireless networks and free-as-in-freedom communication. Thanks to Mark Dailey who taught me to be a hacker and gave me my first real opportunity in industry. Thanks to my very supportive and helpful advisor, Prof. Doug Sicker, and my at-times informal advisor, Prof. Liz Bradley, as well as all of the wonderful teachers I have had along the way. Not least of all, the computer science teachers who got me started on this path: Dennis Heckman, Elizabeth Odekirk-Hash, Dave Blizzard, Melanie Mitchell, and Bart Massey. Finally, but most importantly, thanks to my loving family, wonderful partner Becky, amazing friends, big mountains, fast bicycles, hard climbs, loud music, delicious food, strong ales, and sunny Colorado days, all of which helped keep me sane and positive during the struggle.

Contents

Chapter

1 Introduction 1

1.1 Motivation and Applications . . . 3

1.1.1 Cognitive Networks . . . 3

1.1.2 Self-optimizing Networks . . . 3

1.1.3 General Network Evaluation and Planning . . . 4

1.1.4 Contractual Verification . . . 5

1.1.5 Detecting Spurious Emissions . . . 5

1.2 The Case for Hybridization . . . 6

1.3 The Case for Geostatistics . . . 8

1.4 Thesis Statement . . . 8

1.5 Dissertation Outline . . . 9

2 Path Loss Prediction 11

2.1 Radio Propagation Basics . . . 14

2.1.1 Signal Propagation . . . 14

2.1.2 Path Loss . . . 15

2.2 Modeling Path Loss *A Priori* . . . 18

2.2.1 Theoretical/Foundational Models . . . 20

2.2.2 Basic Models . . . 23

2.2.3 Terrain Models . . . 36
2.2.4 Supplementary Models . . . 39
2.2.5 Stochastic Fading Models . . . 43
2.2.6 Many-Ray Models . . . 44
2.3 Modeling With Measurements . . . 46
2.3.1 Explicit Mapping . . . 47
2.3.2 Partition Models . . . 48
2.3.3 Iterative Heuristic Refinement . . . 48
2.3.4 Active Learning and Geostatistics . . . 49
2.4 Comparative Studies . . . 51
2.5 Discussion . . . 52

3 Bounding the Error of Path Loss Prediction 55
3.1 Measurement . . . 56
3.1.1 Packet-Based Measurements . . . 58
3.1.2 Rural Measurements . . . 60
3.1.3 Urban Measurements . . . 63
3.2 Implementation Details . . . 69
3.2.1 Terrain Databases . . . 70
3.2.2 Corrections for Hata-Okumura . . . 70
3.3 Method . . . 71
3.4 Simple Log-Domain Data Fitting . . . 72
3.5 Performance Metrics for Path Loss Prediction . . . 79
3.5.1 RMSE and SC-RMSE . . . 79
3.5.2 Competitive Success . . . 80
3.5.3 Individual Accuracy Relative to Spread . . . 80
3.5.4 Skewness . . . 81

3.5.5 Rank Correlation . . . 81
3.6 Results . . . 93
3.6.1 Explicit Parameter Fitting . . . 95
3.6.2 Factors Correlated with Error . . . 97
3.6.3 Distance-Hybrid Models . . . 97
3.6.4 Practical Interpretation . . . 101
3.6.5 Miscellaneous Observations . . . 103
3.7 Evaluation of Raytracing Systems . . . 104
3.7.1 Case Study: REMCOM Wireless Insight and Crowd Sourced Building Models . . . 106
3.7.2 Case Study: EDX SignalPro and Rectilinear Building Models . . . 107
3.7.3 Summary of Results . . . 109
3.8 Discussion . . . 111

4 Large Area Coverage Testing 113
4.1 Method . . . 114
4.2 Case Study: Portland, Oregon . . . 117
4.2.1 Method . . . 119
4.2.2 Results . . . 123
4.3 Discussion . . . 128

5 Geostatistical Coverage Mapping 129
5.1 Geostatistics in a Nutshell . . . 129
5.1.1 A Random Field Called 'Z' . . . 130
5.1.2 The Variogram . . . 130
5.1.3 Kriging . . . 133
5.2 Spatial Sampling . . . 134
5.2.1 Classic Sampling . . . 134
5.3 Interpolation . . . 136

5.4 Method . . . 138
5.4.1 Performance Metrics and Measurement . . . 138
5.4.2 Selecting a Sampling Density and Pattern . . . 141
5.4.3 Krige the Residual . . . 145
5.4.4 Variogram Fitting . . . 149
5.4.5 Goodness of Fit . . . 152
5.4.6 Mapping with Ordinary Kriging . . . 154
5.4.7 Visualization . . . 156
5.5 Summary and Conclusion . . . 158

6 Case Studies 162
6.1 Case Study: University of Colorado WiMax . . . 162
6.1.1 Measurement Apparatus and Procedure . . . 164
6.1.2 Comparison of Performance Metrics . . . 167
6.1.3 Possible Sources of Systematic Sampling Error . . . 171
6.1.4 Spatial Data Characterization and Variogram Fitting . . . 173
6.1.5 Mapping with Ordinary Kriging . . . 178
6.2 Map Combining . . . 184
6.2.1 Data Combining . . . 184
6.2.2 Map Combining . . . 184
6.3 Case Study: Verizon LTE Coverage on the University of Colorado Campus . . . 189
6.3.1 Measurement Apparatus and Procedure . . . 189
6.3.2 Performance Metrics . . . 192
6.3.3 Spatial Data Characterization and Variogram Fitting . . . 197
6.3.4 Mapping with Ordinary Kriging . . . 197
6.4 Modeling Small Scale Effects and Deviation from Stationarity . . . 205
6.5 Summary and Conclusions . . . 215

7 Drive-test and Crowd-Sourced Coverage Mapping 219

7.1 Drive Test Coverage Mapping . . . 220

7.1.1 Method . . . 220

7.1.2 Experiment . . . 221

7.1.3 Results . . . 223

7.1.4 Discussion . . . 228

7.2 Crowd-Sourced Coverage Mapping . . . 232

7.2.1 Feasibility Study using Simulated Mobility . . . 232

7.2.2 Case Study: Open Signal Maps . . . 238

7.3 Summary and Conclusion . . . 247

8 Optimized Spatial Sampling 251

8.1 Optimized Sampling . . . 251

8.2 Method and Implementation . . . 254

8.2.1 Specification of Measurement Boundaries . . . 255

8.2.2 Metrics of Sampling Gain . . . 258

8.2.3 Simple Random Sampling . . . 259

8.2.4 Second Phase Greedy Optimized Sampling . . . 260

8.2.5 Second Phase Spatial Simulated Annealing Optimized Sampling . . . 263

8.3 Parallel Spatial Simulated Annealing . . . 273

8.3.1 Experiment 1: N = 10 . . . 277

8.3.2 Experiment 2: N = 25 . . . 277

8.3.3 Experiment 2: N = 50 . . . 277

8.4 Case Studies: University of Colorado WiMax . . . 280

8.4.1 Excluding Null Measurements . . . 289

8.4.2 Phase Three Sample . . . 295

8.5 Summary and Conclusion . . . 295

9 Conclusion 300

Bibliography 306

Appendix

A EDAM: The Effective Directivity Antenna Model 321

A.1 Directional Models . . . 322

A.2 Method . . . 323

A.2.1 Data Collection Procedure . . . 324

A.3 Measurements . . . 325

A.3.1 Experiments Performed . . . 325

A.3.2 Normalization . . . 327

A.3.3 Error Relative to the Reference . . . 327

A.3.4 Observations . . . 330

A.4 A New Model of Directionality . . . 336

A.4.1 Limitations of Orthogonal Models . . . 336

A.4.2 An Integrated Model . . . 339

A.4.3 Describing and Predicting Environments . . . 342

A.5 Simulation Process . . . 344

A.6 Summary and Conclusion . . . 345

B The Stability of The ITM for Typical Problems 347

B.1 Implementation . . . 348

B.2 Experiment . . . 348

B.3 Results . . . 349

B.4 Discussion . . . 349

C Reference Source Code 354

C.1 Kriging Variance 354

C.2 Path Loss Prediction 355

C.3 Effective Signal to Noise Ratio 379

C.4 Spatial Simulated Annealing 384

C.5 Variogram Fitting and Kriging 386

C.6 Anritsu National Instruments Interface 390

Acronyms **393**

Tables

Table

2.1 *A priori* models studied along with their categorization, required input, coverage remarks, relevant citations, and year of (initial) publication. 19

2.2 Commonly used mathematical symbols. 21

3.1 Summary of data sets 57

3.2 Summary of results by data set 78

4.1 Point test state categorization 123

4.2 Random sample performance summary 126

4.3 Summary of a selection of long-term test statistics. 126

5.1 Requirements of an appropriate sampling design 135

5.2 Summary of derived best practices for geostatistical mapping of wireless network coverage. . 161

6.1 Specifications of five University of Colorado WiMax BSs. 164

6.2 Best fit statistics for variogram fitting of CU WiMax BSs. 179

6.3 Specification and location of Verizon LTE cell basestations around Boulder, Colorado. The column N Obs. provides a count of the number of times each BS was observed in our data collection. 190

6.4 Table 1 of best fit statistics for variogram fitting of LTE data. 203

6.5 Table 2 of best fit statistics for variogram fitting of LTE data. 204

7.1 Best fit statistics for variogram fitting of resampled OSM AT&T data. 240

8.1 Summary of random sample validation and home prediction accuracy for phase 2 samples. . 285

8.2 Summary of cross validation and fit-statistics for phase 2 samples. 285

8.3 Summary of random sample validation and home prediction accuracy for phase 2 samples. . 292

8.4 Summary of cross validation and fit-statistics for phase 2 samples. 292

A.1 Summary of data sets. 328

A.2 Factors influencing fitted offset values, 16-bin case. 343

A.3 Summary of Data Derived Simulation Parameters: Gain-offset regression coefficient (K_{gain}), offset residual std. error (S_{off}), and signal strength residual std. error (S_{ss}). 344

Figures

Figure

2.1 Path loss model family tree. Individual models are shown as circles and categories as are shown as rectangles. Major categories are green. Minor categories are blue. 13

2.2 Horizontal and vertical radiation patterns for a (highly directional) 24 dBi parabolic dish antenna. Image taken from L-COM antenna specifications [125]. 15

2.3 Horizontal and vertical radiation patterns for 7 dBi colinear omnidirectional antenna. Image taken from L-COM antenna specifications [125]. 16

2.4 Schematic of link geometry used by basic models. 23

2.5 Schematic of link geometry used by the Flat-Edge family of basic models. 32

2.6 Schematic of link geometry used by terrain models. 37

2.7 Example of sectorized propagation model for a single transmitter using the Robinson model. The measured (oracle) coverage is given as blue circles. The predicted/fitted coverage is given as sector boundaries that are adjusted (pushed and pulled) by additional measurements. Figure taken from [200]. 50

2.8 Example of fitted step function to measurements for the Robinson method. Figure taken from [200]. 50

3.1 Linear fit to RSS error observed from commodity cards during calibration. 58

3.2 The largest of three disconnected sections of the network (80x100km). Link color indicates strength: blue implies strong, red implies weak. Backhaul nodes (mainly 5.8 GHz) are red and CPEs are light blue. 61

3.3 Visual schematic of three urban data sets. A: roof to roof measurements from CU WART (Wide Area Radio Testbed), B: ground (utility poles) to ground (mobile node) measurements in Portland, Oregon, C: roof to ground and ground to roof measurements from CU WART. . 64
3.4 University of Colorado Wide Area Radio Testbed (CU-WART) 64
3.5 Google WiFi Network in Mountain View, California 66
3.6 TFA-Wireless Network measurements in Houston, Texas 68
3.7 Schematic showing the process of parallel computation of path loss predictions using many models and many links. 72
3.8 Explicit power law fits to Data. Fit parameters are provided on the plots. 73
3.9 Explicit power law fits to Data. Fit parameters are provided on the plots. 74
3.10 Number of samples required for naïve fit. Plots show fit standard error for fits increasing random samples and a horizontal line is given at the RMSE obtained for all points. 75
3.11 Number of samples required for naïve fit. Plots show fit standard error for fits increasing random samples and a horizontal line is given at the RMSE obtained for all points. 76
3.12 Schematic explaining error (ϵ) and spread-corrected error (ϵ') in terms of measurement spread and measured and predicted median values. 80
3.13 Five metric results for **WMP** data set 82
3.14 Five metric results for **PDX** data set 83
3.15 Five metric results for **PDX/stumble** data set 84
3.16 Five metric results for **WART** data set 85
3.17 Five metric results for **COST231** data set 86
3.18 Five metric results for **Boulder/ptg** data set 87
3.19 Five metric results for **Boulder/gtp** data set 88
3.20 Five metric results for **Google** data set 89
3.21 Five metric results for **TFA** data set 90
3.22 Five metric results for all **Urban** data sets 91
3.23 Five metric results for all data sets combined 92

3.24 Explicit parameter fitting for the Allsebrook-Parsons and Flat-Edge model parameters. . . . 96

3.25 Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer. 98

3.26 Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer. 99

3.27 Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer. 100

3.28 Comparison of predicted coverage maps for Portland, Oregon using two well-performing models, with and without same scale Gaussian error included. True green indicates predicted recieved signal at -30 dBm and true red indicates predicted received signal the noise floor (-95 dBm). Intermediary values are linearly interpolated between these two color values. . . 102

3.29 Correlation between predicted values and observed values using REMCOM ray-tracing software and WiMax data. The dotted line has a slope equal to 1, which the data points would fall upon if the predictions were perfect. Deviations from this line indicate the magnitude of error. Fit and correlation statistics are given for the aggregate (all APs) predictions. To simplify the plot, points where the prediction software refused to make a prediction have been censored, as well as locations where there was no signal observed. 108

3.30 Correlation between predicted values and observed values using EDX ray-tracing software and WiMax data. The dotted line has slope equal to 1, which the data points would fall upon if the predictions were perfect. Deviations from this line indicate the magnitude of error. Fit and correlation statistics are given for the aggregate (all APs) predictions. To simplify the plot, points where the prediction software refused to make a prediction have been censored, as well as locations where there was no signal observed. 110

4.1 Signal strength from APs in the POC area. Lighter dots (green) indicate stronger signal. . . . 118

4.2 Random locations and their categorization. Green (light grey) dots were tested, purple and orange (grey) were points within the POC that were excluded because they were inaccessible, and red (dark grey) were excluded because they were not within the POC. 120
4.3 Testing apparatus. A battery powered Netgear WGT634u wireless router outfitted with a GPS device, USB storage, speakers, and an enable key. 121
4.4 Contour map of p-values for an exact binomial test as a function of maximal distance to an AP (i.e., only concerning samples within some radius) and hypothesized coverage percent. p-values below $\alpha = 0.05$ reject the null hypothesis that the hypothesized coverage percent is possible given the observations. 125
4.5 Packet loss for long term test at site C as a function of time of day. Measurements are averaged across days and bucketed per hour. 127

5.1 Explanation of variogram model parameters using the Matérn model as an example. Figure taken from [236]. 131
5.2 Examples of sampling schemes 134
5.3 Example of frequency selective fading. Figure taken from [90]. 138
5.4 Example of uniform equilateral triangular sample with $h = 100m$. Markers highlighted blue (as opposed to white) are within 40 wavelengths (approximately 5 meters) of a measurement point. 144
5.5 Path loss measurements for “pdx90” AP, both as-collected and resampled at varying lag distances. Each figure contains four plots which show the spatial and value distribution of the processes 146
5.6 Path loss measurements for “pdx90” AP, both as-collected, and resampled at varying lag distances. Each figure plots the path loss as a function of distance on a log-log plot. A linear least squares regression fit line, and parameters are given. 147
5.7 Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances. . 148
5.8 Empirical semivariograms of path loss for “pdx90” AP without de-trending. 150

5.9 Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with Frii’s freespace model used for de-trending. 151

5.10 Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with offset- and slope-fitted Frii’s freespace model used for de-trending. Measurements at the noise floor have been inserted at points where an observation was unable to be made. . . 153

5.11 Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with Frii’s freespace model used for de-trending. Measurements at the noise floor have been inserted at points where an observation was unable to be made. 155

5.12 Comparison of color maps using a Kriged map of WiMax CINR coverage for one BS at the University of Colorado. 157

5.13 Examples of coverage map (for CU WiMax cuEN node) overlayed on Google Earth orthoimagery, digital terrain, and 3D models. 159

5.14 Examples of coverage map (for CU WiMax cuEN node) overlayed on Google Earth orthoimagery, digital terrain, and 3D models with measurement locations and values. 160

6.1 Map of University of Colorado and 100m uniform equalateral triangular sample. Measurements are limited to the main campus, which is outlined in red. 163

6.2 Diagram showing connectivity and specification of WiMax measurement cart devices. 165

6.3 Diagram of WiMax measurement cart. 165

6.4 Correlation between various metrics. 169

6.5 Correlation between WiMax throughput and metrics that are meaningfully correlated with application-layer performance. 170

6.6 Correlation between CINR and GPS accuracy 172

6.7 Measurement location error and GPS accuracy as a function of the measurerer. 174

6.8 Measurements for CU WiMax cuEN (GENI) node. 175

6.9 De-trending fits for the CU WiMax cuEN (GENI) node. Only the metrics that can be converted to path loss and de-trended (i.e., SNR and equivalents) are shown. 176

6.10 Empirical variogram and fits of four metrics for CU WiMax cuEN (GENI) node. 177
6.11 Maps for cuEN node. The left maps show the excess (residual after trend is removed). The center maps show the re-trended signal map. The right maps show the residual kriging variance of the other maps. 181
6.12 Maps for cuGE node. The left maps show the excess (residual after trend is removed). The center maps show the re-trended signal map. The right maps show the residual kriging variance of the other maps. 182
6.13 Binary coverage maps with the threshold of CINR=30 183
6.14 Empirical variogram and fits of four metrics for the combined CU WiMax measurements. . . 185
6.15 Kriged maps for combined CU WiMax measurements using the CINR metric. 185
6.16 Binary coverage map for the combined CU WiMax measurements. 185
6.17 Kriged maps for combined CU WiMax measurements using the CINR metric. 188
6.18 Boolean threshold-based maps for map-combined CU WiMax measurements. The CINR map uses a threshold of 40 dB and the ESNR maps use a threshold of 20 dB. 188
6.19 Diagram showing connectivity and specification of LTE measurement cart. 190
6.20 Diagram of LTE measurement cart. 191
6.21 Correlation between upstream throughput performance and physical-layer metrics. 195
6.22 Correlation between physical-layer metrics. 196
6.23 Log/log fit of path loss for LTE BS 369. 198
6.24 LTE Measurements. 199
6.25 Measurements for LTE BS 369. 200
6.26 Variogram fits for all measurements combined. 201
6.27 Variogram fits for BS 369. 202
6.28 Kriged maps for BS 369. 206
6.29 Kriged maps for all measurements combined. 207
6.30 Map-combined maps for all measurements combined. 208
6.31 Threshold-based map-combined maps for all measurements combined. 209

6.32 Distribution of spread and comparative QQ-plots for measurements taken in within 40 wavelengths of each other (i.e., clustered) versus at the same point at different times (i.e., unclustered) for two different definitions of measurement spread (MAD and range). 211

6.33 Amount of spread (variation) as a function of time elapsed between measurements using two different metrics. 212

6.34 Comparison of mean absolute deviation for metrics both in the same place at different times (column FALSE, i.e. unclustered) and at different places and different times, but less than 40 wavelengths apart (column TRUE, i.e., clustered). 214

6.35 Mean absolute deviation as a function of time elapsed between measurements in seconds. There appears to be no discernable correlation. 216

6.36 Number of samples per bucket for time elapsed between repeated measurements. 217

7.1 Schematic describing validation process for drive-test coverage mapping. 222

7.2 Performance results for different resampling lags and careful and aggressive resampling using four metrics of interest. 225

7.3 ROC and DET curves for each dataset using “aggressive” resampling. 226

7.4 Hole prediction accuracy as a function of sampling density. 227

7.5 Map-combined (maximum combining) Kriged coverage map for PDX Data using best variogram and h=100m. 229

7.6 Map-combined (maximum combining) Kriged coverage map for Google Data using best variogram and h=100m. 230

7.7 Map-combined (maximum combining) Kriged coverage map for TFA Data using best variogram and h=100m. 231

7.8 Sample coverage as a function of percentage of participating users and length of experiment. The left-hand plots show the number of visits to each 5 meter cell on a log scale. The right-hand plots show a boolean map of cell coverage. 235

7.9 Sample coverage as a function of percentage of participating users and length of experiment. 236

7.10 Open Signal Maps measurements for AT&T network overlayed on Google orthoimagry. . . . 239
7.11 Performance results for OSM data at different resampling lags using four metrics of interest. 241
7.12 Performance results for OSM data at different resampling lags using four metrics of interest. 242
7.13 Performance results for OSM data at different resampling lags using four metrics of interest. 243
7.14 Performance results for OSM data at different resampling lags using four metrics of interest. 244
7.15 Map-combined maps using maximum-based combining for OSM data. 245
7.16 Map-combined threshold maps using maximum-based combining for OSM data. 246
7.17 Performance results for OSM data at different resampling lags using four metrics of interest. 248
7.18 ROC and DET curves for OSM measurements using “aggressive” resampling. 249
7.19 Hole prediction accuracy as a function of sampling density for OSM data. 249

8.1 Systematic (grid) sampling augmented with nested ”random” samples. In the NxN case, points are placed on an equally spaced regular grid of NxN points and then N additional points are sampled in the immediate neighborhood of N grid points. This has the effect of creating an initial sample that both covers the region and has a range of lag distances between measurements for variogram estimation. Figure taken from [63]. 253
8.2 Optimized placement of second-phase samples according to the weighted Kriging variance proposed by Delmelle. Figure taken from [63]. 253
8.3 Convex hull around first-phase sample points for cuEN node. 256
8.4 Manual specification of boundary and unmeasureable polygons using the Quantum GIS software. 257
8.5 Performance gain from increasingly large simple random samples. 261
8.6 Maps used for sample optimization for cuEN node. 262
8.7 Improvement (gain) from iterative greedy sampling using the WPE map. 262
8.8 Greedy sample of 100 points. Preexisting sample locations are denoted with circles and second-phase (greedy) samples are filled (red) triangles. 264
8.9 Metropolis probability plot for spatial simulated annealing. 266

8.10 Improvement (gain) from spatial simulated annealing using the WPE map for sets of 10 or 25 additional points. 268
8.11 Optimized sample for 10 and 25 points after 1000 iterations. 269
8.12 Improvement (gain) from spatial simulated annealing using the WPE map for 50 additional points with increasing numbers of iterations 270
8.13 Optimized sample using the WPE map for 50 additional points with increasing numbers of iterations 271
8.14 Optimized sample using the WPE map for 50 additional points with increasing numbers of iterations, but with the ESNR metrics. 272
8.15 Optimization timeseries for $N = 25$. 275
8.16 Distribution of gain for $N = 25$. Register your sighting of Thesis Waldo at thesiswaldo.org. Waldo image by Martin Handford, distributed by Classic Media Distribution Ltd.. 276
8.17 Parallel versus sequential performance for $N = 10$. 278
8.18 Parallel versus sequential performance for $N = 25$. 279
8.19 Distribution of gain for $N = 50$ with temperatures of 1000 and 2000. 281
8.20 Parallel versus sequential performance for $N = 50$. 282
8.21 Optimized second-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery. 284
8.22 Comparison of second-phase threshold maps for CINR=20. 286
8.23 Comparison of second-phase coverage. 287
8.24 Comparison of second-phase variograms. 288
8.25 Measured second-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery. 290
8.26 Phase-2 Optimized Sample. 291
8.27 Comparison of second-phase threshold maps (threshold is CINR = 40 dB). 293
8.28 Comparison of second-phase coverage. 294
8.29 Phase-3 Optimized Sample. 296

8.30 Measured third-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery. 297

8.31 Comparison of second-phase threshold maps (threshold is CINR = 40 dB). 297

8.32 Comparison of second-phase coverage. 298

A.1 Probability Density Function (PDF) of percentage of dropped measurement packets in a given angle for all angles and all data sets. 331

A.2 Receiver side of measurement setup in floodplain 331

A.3 Floorplan of office building used in Array-Indoor-A, Array-Indoor-B, Patch-Indoor-B, Patch-Indoor-C, Parabolic-Indoor-B, and Parabolic-IndoorC. 332

A.4 Comparison of signal strength patterns across different environments and antennas. 333

A.5 Probability Density Functions (PDFs) for the averaged error process (combined across multiple traces) for each antenna type. 334

A.6 Cumulative Density Functions (CDFs) for the error process (combined across multiple traces) for each antenna type. 334

A.7 Probability Density Function (PDF) of p-values from testing the normality of the error process in each direction for each data set. In all cases, the null hypothesis (that the samples are normally distributed) can be confidently rejected. 335

A.8 Heatmap of p-values for the Mann-Whitney U-test which was run pairwise against the error from the reference pattern in each angle. This plot, which is for Patch-Indoor-A, was chosen as a representative. *All traces showed similar trends*. Darker values indicate very small p-values, meaning that the null hypothesis can be rejected with confidence. In this case, the null hypothesis is that the samples come from the same distribution. The Patch reference pattern is provided on the left for reference. 335

A.9 Differences between the orthogonal model and observed data in dB: $\hat{P_{rx}} - P_{rx}$. 337

A.10 Mean error of orthogonal model for each observation point in the Array-Outdoor-A data set. The format is the same as in figure A.9. 338

A.11 Effect of increasing bin count (decreasing bin size) on modeling precision. 341

A.12 Residual error of the discrete offset model with 16 bins. 341

B.1 Box and whiskers plot of error as a function of precision. 351

B.2 Box and whiskers plot of error as a function of precision, showing only results for single-precision and greater arithmetic. 352

B.3 Running time of ITM algorithm as a function of precision. The 0-bit case is the machine-precision reference implementation. 353

Chapter 1

Introduction

In the last hundred years, wireless transmission has revolutionized the way we communicate. In the first half of the last century, broadcast radio transmission completely changed the way information was distributed on a global scale. In the second half of the century, wireless networks greatly lowered the barrier to information sharing between computer systems. Perhaps because of the level of integration they have obtained in our day-to-day lives, today's wireless networks are simultaneously amazing and disappointing. They allow us to do something that seems almost unbelievable: effortlessly moving bits from one computer to another through the air. Yet, they never seem to work quite as well as we would like in the places we would most like them to.

This thesis focuses on a specific problem that is at the center of many other problems with wireless networks: *there are not good methods for determining how well a given network works over a given area and presenting this information in a meaningful way*. The seemingly fundamental task of drawing a meaningful and accurate picture of the "usable" coverage of an existing wireless network is an open question that this thesis will address.

Solving the general coverage problem involves advancing the state-of-the-art in four integral sub-problems:

(1) **Prediction**: How can the signal quality at a given point (or many points) be predicted if only information about the environment and the transmitter is known?

(2) **Measurement**: Assuming measurements are to be made to correct or evaluate a model, how and

where should they be made? How many are necessary? What are the tradeoffs in terms of cost and accuracy for these measurements?

(3) **Interpolation**: When measurements are in hand, how can they be used to make inferences about the coverage at locations where measurements have not been made? Which of the numerous interpolation strategies is most appropriate for mapping wireless network coverage?

(4) **Presentation**: How can the resulting maps and spatial processes be presented in a clear and meaningful way, while not over- or under-estimating the network's abilities? How does the ideal visualization differ for end-users, designers, and operators of these networks?

To provide answers to all of these questions is a substantial undertaking. However, this thesis makes strides in each subproblem. First, to understand the practical accuracy of existing methods, a great number of prior models have been carefully analyzed, implemented, and performance-tested. Measurement methodologies are advanced by applying statistically robust, domain-appropriate spatial sampling methodologies. Methods for optimizing multi-phase sampling schemes are developed so that those measurements that are made have been placed to enable the largest gains with the least amount of work. This thesis proposes the use of geostatistical interpolation and modeling for wireless coverage mapping, in large part because this method embraces the intrinsic variability of the radio environment and allows for residual error and variance to be modeled explicitly. These techniques are adapted as necessary and applied to the problem through two novel case studies, and a set of best practices are extrapolated from the lessons learned. Finally, a simple visualization scheme is developed that presents wireless coverage using a color-mapping scheme adapted from medical imaging, and can be interactively used with popular mapping software. By providing a complete method for measurement-based coverage mapping, this thesis seeks to provide a novel, systematic, well-defined, and thoroughly evaluated approach to the very important problem of measurement-based coverage mapping.

1.1 Motivation and Applications

A general solution to the coverage mapping problem could have substantial impact on both current and next generation wireless technologies. This section briefly discusses five important applications of measurement-based coverage mapping.

1.1.1 Cognitive Networks

Cognitive networks are considered by many to be the next step in intelligent spectrum use [145]. Although much of the available (useful) wireless spectrum has been auctioned to particular primary users, there has been a great deal of work showing that not all of this spectrum is being fully utilized. Indeed, there are large areas distributed in space and time that can be gleaned by a secondary user for communication (e.g., [229]). However, for the cognitive radio model to work, accurate coverage maps, or "radio environment maps", are necessary to provide insight into locating and avoiding primary users. Some proposals suggest that predictive models be used to estimate transmission boundaries (e.g., [89]). However, as will be shown in chapter 3, the error associated with these models can be substantial and unpredictable. More recently, some researchers are investigating the possibility of measurement-based mapping in this domain using fixed or mobile sensors (e.g., [195, 56, 81]), but there are still many open questions. This thesis makes some first steps to answer some of them. Bounding the error of predictive models will help motivate a measurement-based solution. Then, when studying the effect of spatial sampling strategies, optimized sampling provides insight into where sensors must be deployed to create accurate maps of existing usage, to determine how many sensors are necessary, and to establish whether user-collected (i.e., "crowd-sourced") data is useful for generating maps.

1.1.2 Self-optimizing Networks

Networks that make decisions about their channel usage and other configurations benefit from accurate information about the channel in terms of both current coverage and interference. For instance, in [116], Kanade *et al.* propose a network optimization strategy where routing decisions are made based upon

inferences of link quality taken from a measured signal map, which they call the “wireless manifold”. Understanding how and where (and even when) measurements must be made to model the Radio Frequency (RF) environment with sufficient precision is an an important open question that underlies these proposals. The robust geostatistical approaches developed in this thesis begin to answer these questions. In an environment with highly mobile transmitters, where the surrounding RF environment cannot be assumed to be optimal, optimized multi-phase sampling of the sort desribed in chapter 8 might also be used to iteratively learn and correct a coverage map over time.

1.1.3 General Network Evaluation and Planning

Network operators require an understanding of the extent of coverage of wireless networks in such a way that it can be used to repair problems (holes), expand the network, and communicate coverage to users and marketers. By identifying areas of potential inter-node interference or coverage gaps, a network operator can choose to tune the antenna orientation and tilt of a given Base Station (BS) antenna, or add transmitters where they are needed. For instance, by identifying the coverage holes in microcell outdoor networks, a cell network operator might choose where nanocells could be installed to address local regions of poor connectivity.

Coverage maps can also be used in the planning and build-out phases of wireless network deployment. For instance, in [95], Hills discusses the wireless network at Carnegie Mellon University (CMU) and argues for an iterative deployment process where coverage testing feeds back into deployment decisions. In [115], the authors propose an Access Point (AP)-placement algorithm which uses ray-optical measurements as input. In practice, network operators often obtain information about coverage of their network by collecting data with mobile vehicles, colloquially called “drive-testing” or “war-driving”. For instance, in [43], Byers and Kormann provide a good overview of AP mapping and in [97], geography researchers provide their mapping technique for the unplanned networks of Salt Lake City, Utah. Despite the prevalence of this technique, seldom are important concerns such as sampling bias, completeness, choice of performance metric, or statistical significance considered. This thesis investigates methods for principled spatial sampling in wireless coverage mapping, and appropriate interpolation techniques, which can provide insights into how

"drive-testing" and investigative coverage mapping (for planning or diagnosing problems) might be made more robust with the use of appropriate statistical techniques.

1.1.4 Contractual Verification

Being able to make a strong statement about the extent of coverage for a wireless network is crucial for verifying contractual requirements for network deployments. In a typical scenario a contract will be drafted between the client wishing to build a wireless network, and the company they have hired to build it. In this contract, the goal coverage criterion can be specified along with an appropriate testing methodology to determine when the contractual obligations have been fulfilled. Chapter 4 will look at a municipal wireless network in Portland, Oregon that was substantially harmed by the operators' inability to identify coverage gaps [121, 164, 232]. In the business of wireless telephones, communicating wireless coverage to end users has become an important business practice, which educated consumers use to choose their provider [227, 140, 31].

There are no shortage of companies, both big and small, that offer contractual coverage testing as a service (e.g., [239, 4]). However, there are no universal standards or best practices for how coverage testing should be performed, and in some cases the techniques used by consulting firms may lack statistical or procedural rigor. Due to variation in the methods, results from tests might not be comparable or reproducible. Section 4 will look at the coverage testing problem in the context of municipal wireless contractual verification. By proposing a straightforward and robust method for coverage testing, hopefully the uncertainty in this domain can be mitigated. In some networks, contractual verification may involve more rigid specifications including varying coverage requirements in different regions. The coverage mapping methods described in chapter 5 can help to address these issues.

1.1.5 Detecting Spurious Emissions

A related problem involves detecting spurious emissions. In some scenarios, regulatory enforcement agencies may be required to determine whether rogue transmitters are creating harmful interference or operating outside of their band. For instance, in a recent decision, the Federal Communications Commission

(FCC) determined that emissions from a neighboring commercial network might impact Global Positioning System (GPS) devices [216]. In this scenario, the same coverage mapping problem presents itself. The *de facto* approach used today in this scenario involves point-testing and *predicting* out-of-band emissions using a model. A generalized coverage mapping method, adapted to making inferences about the RF environment could be useful in this scenario to determine not only whether spurious emssions are present, but also the source and exact extent of interference.

1.2 The Case for Hybridization

There are two approaches to coverage mapping that dominate the state of the art today: direct measurement and *a priori* modeling. It stands to reason that these themes would prevail, as they comport to Occam's razor, but *how well do they work*?

Direct measurement is straightforward: visit a large number of points in the area of interest, measure the signal strength at those points (and ideally perform higher-layer tests as well), and then use these measurements to draw a coverage map. The problem with this approach is that it scales poorly and becomes stale quickly. Exhaustive measurement is very laborious for small networks, and for networks that are city-sized, it is likely cost prohibitive. When measuring a large area, one must choose a subset of points to measure, and appropriate sampling strategies are not well known, nor is it well understood what sources of bias can stem from inappropriate sampling. For instance, many cell carriers typically rely on "drive-test" measurements, where measurements are made exclusively with a mobile (vehicular) tester along streets, without much concern for how the atypical propagation environment created by streets, as well as sampling bias, may effect the validity of measurements. Once the data is collected, it is not clear how well this data will age. There can be substantial small-scale and large-scale temporal variation in the radio channel. At what point does it become too stale to use? How often should it be updated?

Another problem with this approach is that the best way to interpolate between measurement points is not well known, nor are there standards for how to present this data in a useful way. Linearly (or exponentially) interpolating between neighboring measurements causes an uncertain amount of error as a result of this smoothing. How does one make a map that is actually meaningful for the person using it? Mapping

schemes that only plot the expected received signal strength (or signal to noise ratio) neglect a number of channel attributes that may significantly contribute to the usability of a network at that point, such as neighboring interference or link asymmetry. In short, despite its attractive simplicity, direct measurement alone is not enough—it does not properly treat the problems of sampling, interpolation, and presentation.

The other popular way to make a map is using a predictive model. This is especially useful for planning networks that are not yet built, and hence, cannot be directly measured. Maxwell's equations describe the propagation of plane waves. It is well known that signal degrades approximately proportionaly to the square of the distance. Is it possible, then, to use an analytical model to predict the coverage in a given environment *a priori*? There is no shortage of existing models that try to predict signal attenuation as a function of distance between points, using any number of other variables and parameters. These models come in every shape and size, but it is not known how well they work in general. Certainly, picking a propagation model from the literature and using it to create a coverage map is one approach. However, saying how accurate such a map is without making direct measurements is impossible. And, choosing incorrectly can have penalties whose severity cannot be determined *a priori*. For instance, in [44], Camp *et al.* show that small changes in model parameters used for planning a wireless mesh network can result in massively under- or over-provisioned networks. Chapter 3 will attempt to define bounds for the error associated with these *a priori* models in practical applications, and show that on their own, they cannot sufficiently model the channel. There is no silver bullet here, either.

Because neither approach works well enough on its own, the approach investigated here is a hybridization of these two: measurement-corrected modeling. The starting assumption is that creating an accurate coverage map for a given network requires some direct measurement. However, because measurement is costly, it is desireable to make *as few measurements as possible to generate a map within application-appropriate accuracy bounds*. To address this problem, this thesis will look to the geostatistics literature, which has much to say about rigorously sampling random fields. Chapters 5, 6, and 7 provide an adapatation of these methods to the coverage mapping problem. Then, chapter 8 will suggest optimized sampling strategies to generate a single system that is able to create a more accurate coverage map than is possible with either approach alone, while requiring less work than an exhaustive measurement campaign.

1.3 The Case for Geostatistics

In 1951, Daine Krige revolutionized the field of mine valuation by developing new statistical methods for spatial processes. Mine valuation is the task of determining the grade of ore or the amount of precious metal over some region. Predicting the shape and distribution of this field is essential to planning mines, and hence there are substantial penalties for inaccuracy. Prior to his proposals, the task of mapping the grade of ore in mines was in a similar state to how things are today with mapping the coverage of wireless networks. It is well known that creating maps is necessary, but there is little consensus about the best way to do it, and sources of measurement bias and modeling error are not well understood. In [123], Krige made the case for statistics:

> The need for greater uniformity in valuation procedures, and for the limitation as far as possible of the personal element cannot be disputed. The solution to this problem lies, in my opinion, in the extensive application of statistics. I do not wish to imply, however, that statistics is a miracle tool with rigid procedures that can be applied indiscriminately on any mine without a proper appreciation of local conditions. On the contrary, a clear concept of the problems involved is essential, and this can emanate only from practical experience. Once the necessary spade work has been done, however, the routine application of statistics on any mine will involve only simply arithmetical calculations well within the scope of the average surveyor and sampler.

This thesis will take a “Krigian” approach. A fundamental assumption here is that the solution to the coverage mapping problem is, like mine valuation, a task involving costly sampling of points in a random field and using these samples to infer the shape of the field overall. However, as Krige eloquently points out, reckless application of statistical methods here leaves us no better off than we started—it is necessary to marry practical knowledge about how wireless networks work with proper statistical methods. In the end, as Krige did with mine valuation, this thesis will provide a complete method that brings the currently costly and complex task of coverage mapping into the domain of the “average surveyor”.

1.4 Thesis Statement

The following thesis is asserted:

> Domain-appropriate geostatistical methods can provide a solution to wireless coverage mapping that (a) is more accurate than is possible with *a priori* modeling approaches and (b) requires fewer measurements than explicit, undirected measurement-based approaches.

Chapter 3 will show that the minimum practical accuracy of existing approaches to *a priori* modeling is on the order of 9 to 12 dB. Chapters 5 and 6 present a geostatistical method for wireless coverage mapping which more than halves this error in practice. Chapter 7 shows that this method can produce results that compete with the state-of-the-art methods for finding coverage holes using drive-test measurements, both in terms of minimizing the number of measurements and predictive accuracy. Finally, chapter, 8 will extend the methodology with a iterative optimized sampling system that can further tune coverage maps with intelligently placed second-phase samples.

Because the landscape of all wireless networks is large and varied, this evaluation focuses on a type of network of particular interest: large-scale, microcell, outdoor, urban wireless networks operating in the UHF band (300 MHz to 3 GHz). This sort of network is widely used for providing wireless network access to cities and towns (e.g., [146]), university campuses, mobile cell networks, commercial and industrial campuses, and military sites (e.g., [95]). Because of their relatively low cost and accessibility, these networks have also gained some traction in sparse rural applications (e.g., [9, 180]). Besides existing networks, the results here are applicable to future networks that operate using a similar microcell design in urban areas and at similar frequencies. And, as metrics and models are improved with time, they can be "plugged in" to the methods proposed here. Although these sorts of networks are the focus of this thesis, the methods developed are likely applicable to a large class of similar networks and frequencies.

1.5 Dissertation Outline

The next chapter provides background on the problem, including a discussion of practical wireless path loss prediction, and describes the state of the art in terms of coverage mapping. Chapter 3 puts the state of the art to the test by performing a rigorous evaluation of 30 analytical and empirical propagation models from the literature and puts practical bounds on the prediction error of these models with respect to ground-truth data. These results show that "simple" approaches to path loss prediction are troublesome

and choosing amongst them is precarious. This observation motivates the principled measurement-based coverage mapping method developed in the subsequent chapters. Chapter 4 begins with the smaller problem of coverage *testing*, where the goal is to determine the *percentage* of area covered at some level of performance. These methods are applied to the problem of testing the coverage of a large municipal wireless network in Portland, Oregon. Chapter 5 introduces the core proposal of the thesis, which provides a geostatistical method for sampling, measurement, and coverage *mapping*. The efficacy of this method is evaluated through two case studies provided in chapter 6, which seek to map the coverage of WiMax and Long Term Evolution (LTE) networks in Boulder, Colorado. As a practical extension to this proposal, chapter 7 analyzes re-sampling approaches which allow for commonly available drive-test data to be used with the geostatistical methods proposed here. Chapter 7 also introduces the topic of crowd-sourced coverage mapping where end-users volunteer to collect measurements of network coverage. A feasibility study is performed using a mobility model, as well as a case study using data collected with smart phones in greater Colorado by the company Open Signal Maps. Chapter 8 presents a method to fine-tune maps generated with this system using iterative refinement with optimized multi-phase sampling. Finally, chapter 9 will conclude the thesis with a recap of results and a discussion of possible future directions.

The main text is accompanied by several appendices which provide results from related experiments. Appendix A describes the Effective Directivity Antenna Model (EDAM), a path loss model for directional antenna systems. Appendix B describes an experiment seeking to understand the numerical stability of path loss models that make a large number of numerical approximations using the Uniform Theory of Diffraction (UTD). And, appendix C provides source-code listings for the most important algorithms produced by this thesis.

Chapter 2

Path Loss Prediction

Today, wireless networks are absolutely ubiquitous and the importance of their role in our daily lives cannot be underestimated. To a large extent, our ability to build and understand these networks hinges on understanding how wireless signals are attenuated over distance in realistic environments. By predicting the attenuation of a radio signal, we can better plan and diagnose networks as well as build futuristic networks that adapt to the spatiotemporal radio environment. For instance, today's network engineers need methods for accurately mapping the extent of coverage of existing and planned networks, yet the efficacy of those approaches is determined by the predictive power of the underlying path loss model (or interpolation regime). Similarly, researchers who investigate dynamic spectrum access networks require accurate radio environment maps to make appropriate and timely frequency allocation decisions, yet the performance of these systems is tied intimately to their ability to make meaningful predictions about the current and future occupancy of the radio channel.

Since the 1940's, researchers and engineers have pondered this problem and have developed myriad schemes that purport to *predict* the value or distribution of signal attenuation (path loss) in many different environments and at different frequencies. This chapter will attempt to give a complete review of the work to date, updating and extending a series of excellent-but-dated surveys from the last 15 years (e.g., [107, 34, 183, 206, 151]).

The first section provides a high-level tutorial on radio propagation, which may be supplemented with either of the excellent surveys by Rappaport [183] and Seybold [206]. After this introduction, the remaining

[0] Work in this chapter has appeared in [175].

sections provide an exhaustive survey of the deep literature available on path loss prediction methods. The chapter concludes with an overview of the state-of-the-art in path loss prediction and coverage mapping methods, including those that use computationally intense ray-optical calculations, and those that utilize some number of measurements to correct predictions. To make sense of all this prior work, these proposals are organized into a new taxonomy for path loss models that groups them into seven major categories and fourteen subcategories. The seven major categories are:

(1) Theoretical/Foundational Models (§2.2.1)

(2) Basic Models (§2.2.2)

(3) Terrain Models (§2.2.3)

(4) Supplementary Models (§2.2.4)

(5) Stochastic Fading Models (§2.2.5)

(6) Many-Ray (Ray-Tracing) Models (§2.2.6)

(7) Active Measurement Models (§2.3)

The discussion here is exhaustive, including more than 50 proposed models from the last 60 years, 30 of which are described in detail. The models are described at a high level with a brief focus on identifying their chief differences from other models. Figure 2.1 provides a family tree of the majority of path loss models discussed in the following subsections and may prove useful for understanding the lineage of various proposals as well as their functional relationship to one another.

The next section provides a brief tutorial on radio propagation. Section 2.2 discusses the bulk of models, which make their predictions *a priori*, without insight from measurements. Section 2.3 discusses models and methods that do use (possibly directed) measurements to inform their predictions. The final sections conclude with a discussion of opportune areas for future work, and motivation for the next chapter, which will seek to understand the practical error of the most popular of these models in realistic environments.

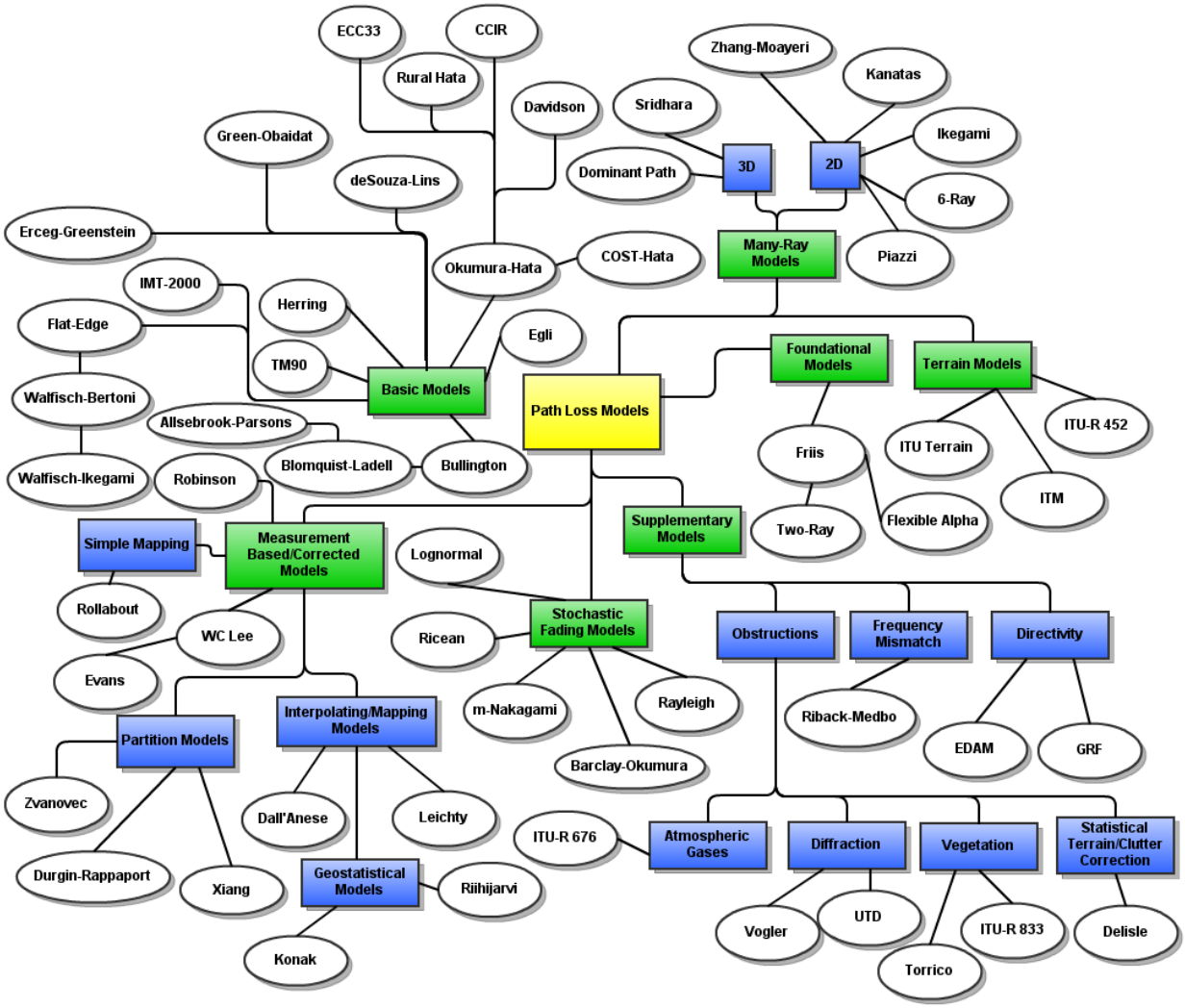


Figure 2.1: Path loss model family tree. Individual models are shown as circles and categories as are shown as rectangles. Major categories are green. Minor categories are blue.

2.1 Radio Propagation Basics

This section introduces the basic concepts of radio propagation. For a more thorough treatment, the intrepid reader can refer to any number of textbooks, including the excellent surveys by Rappaport [183] and Seybold [206].

2.1.1 Signal Propagation

When asked to describe radio, Albert Einstein famously responded:

> You see, wire telegraph is a kind of a very, very long cat. You pull his tail in New York and his head is meowing in Los Angeles. Do you understand this? And radio operates exactly the same way: you send signals here, they receive them there. The only difference is that there is no cat.

The study of radio propagation is largely concerned with what happens in between the head and the tail of the "no cat", so to speak. At each end of the radio link, there is a transceiver that is attached to an antenna of some geometry. The transmitter produces a signal (an electromagnetic plane wave) that is modulated onto the carrier frequency. On its way to the receiver (at roughly the speed of light), the signal reacts with any number of obstacles and then is induced on the receiver's antenna and demodulated. Obstacles in the environment cause the signal to be reflected, refracted, or diffracted, which attenuate the power of the signal (through absorption) and cause scattering and secondary waves. Obstacles that are near the line of sight (LOS) path are said to obstruct the Fresnel zone (technically, the first Fresnel zone's circular aperture) and are most problematic.

In reality it is slightly more complicated than this. Because an antenna radiates its signal simultaneously in all directions, the signal can take many paths to the receiver. Each path may interact with the environment in a chaotically different way and arrive at the receiver delayed by some amount. If these delayed signals are in phase with one another, then they produce constructive interference. If they are out of phase with one another, they produce destructive interference. The spread of this delay is called the *delay spread* and the resulting attenuation is called *multipath fading*. When this attenuation is caused by large

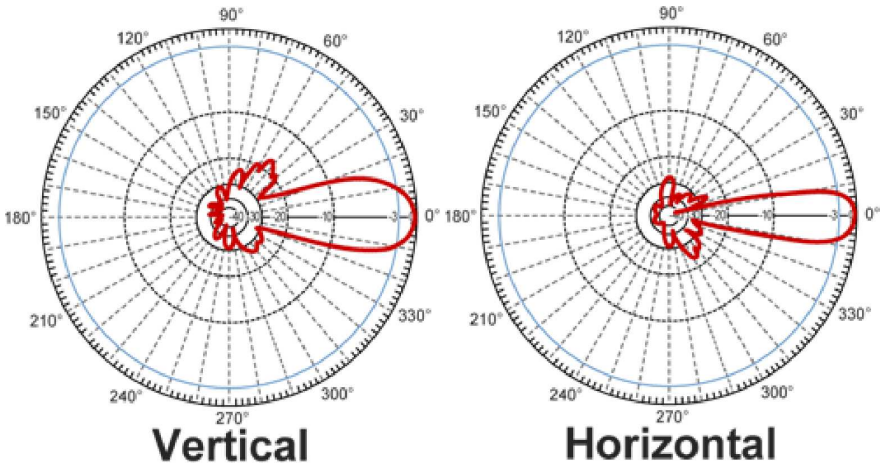


Figure 2.2: Horizontal and vertical radiation patterns for a (highly directional) 24 dBi parabolic dish antenna. Image taken from L-COM antenna specifications [125].

unmoving obstacles it is referred to as *shadowing*, *slow-fading*, or *large-scale fading*. When it is caused by small transient obstacles, and varies with time, it is called *scattering*, *fast fading*, or *small-scale fading*.

When the signals interact with the environment, they can be delayed by reflections, or frequency-shifted by diffractions. Mobile transceivers also incur frequency shift due to Doppler spreading. Frequency shifts and delay spread both contribute to small scale fading.

2.1.2 Path Loss

The *geometry* of the antennas that the transmitter and receiver use emphasizes signals arriving from some directions over others. An omnidirectional antenna emphasizes signals in the azimuthal plane and de-emphasizes signals arriving from above or below. As a result, the gain pattern tends to be shaped like a donut, as can been seen in figure 2.3. A directional antenna, such as a patch panel, parabolic dish, or sector, typically emphasizes signals arriving from a single direction (*lobe*) within some *beamwidth*. The gain pattern of these antennas more closely resembles a baseball bat, as can be seen in figure 2.2. However, perfect isolation is impossible and geometries that emphasize a single direction also have substantial gain in other directions (*side lobes* and *back lobes*) as a result. Antenna gain is typically measured in dBi, which is decibels relative to an isotropic transmitter (an isotropic transmitter's gain pattern is a sphere).

If the transmitter's radio has a transmit power of P_{tx} Watts (W) and an antenna gain of G_t dBi, then

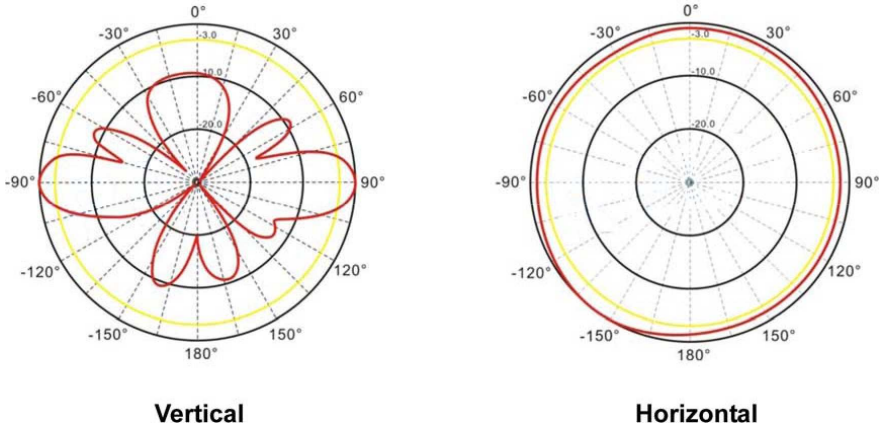


Figure 2.3: Horizontal and vertical radiation patterns for 7 dBi colinear omnidirectional antenna. Image taken from L-COM antenna specifications [125].

the total Effective Isotropic Radiated Power (EIRP) is $P_{tx} * G_{tx}$. In the log domain, P_{tx} is given in dBm, which is decibels relative to a mW, and the EIRP is simply $P_{tx} + G_{tx}$. The entire radio link can then be summarized by the common log-domain link budget equation:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - PL \tag{2.1}$$

with P_{rx} and G_{rx} being the power received at the receiver and the receiver's antenna gain in the direction of the transmitter. Here, the PL term includes all attenuation due to path loss. This formula describes the aggregate gain and attenuation of many competing signals. It also assumes that our radio link is isolated from any sources of external noise in the environment (i.e., thermal noise and interference from other transmitters). Commonly, the signal quality at a given point is written as the ratio between Signal and Noise: $SNR = P_{rx} - N$ (in the log domain). Alternately, if interference from a known set of interferers is included, the Signal to Interference and Noise Ratio (SINR) is defined as:

$$SINR = P_{rx} - \left(N + \sum_{j}^{n} I_j \right) \tag{2.2}$$

For a given receiver design and modulation scheme, there is a known relationship between Signal to Noise Ratio (SNR) and bit error rate. Using this relationship, we can determine the minimum detectable signal for a given radio as a function of the acceptable error rate: $MDS(P_e)$, where P_e is the probability of bit error. Then, determining the points that are covered is simply the set of receiver locations that satisfy the inequality:

$$P_{tx} + G_{tx} + G_{rx} - PL \geq MDS(P_e) \tag{2.3}$$

Because the P and G terms are known for a given link, the difficulty becomes predicting the quantity PL given what we know about the environment and the radio link. In the case of measurement-based approaches, the complementary problem involves interpolating the PL value for the points we have not measured.

As it is defined here, a model's task is to predict the value of $L_t + L_s$ in this log-domain equation:

$$PL = L_t + L_s + L_f(t) \tag{2.4}$$

where L_t is the trivial free-space path loss, L_s is the loss due to shadowing (slow fading) from large unmoving obstacles like mountains and buildings, and $L_f(t)$ is the small-scale fast fading due to destructive interference from multipath effects and small scatterers (which varies with time t). Small-scale fading is often both time and frequency selective, meaning that it varies with time and frequency. Models cannot, without perfect knowledge of the environment, be expected to predict the quantity $L_f(t)$. In most applications, this additional error is computed "stochastically" using a probability distribution (often Raleigh, although Ricean and m-Nakagami are popular). In this way, frequency and time selective fades can be simulated, if not predicted exactly, which allows for the analysis of their effect on modulation schemes (e.g., [90, 213]). In the following sections, the many methods proposed for predicting the value of $L_t + L_s$ and the distribution of $L_f(t)$ are discussed.

2.2 Modeling Path Loss *A Priori*

The models discussed in this section are *a priori*, meaning they make predictions using only available prior knowledge and do not use explicit measurements in their predictions. Hence, these models are most appropriate for making predictions in situations where it is impossible or difficult to obtain measurements. We subdivide these models into six categories:

(1) Theoretical/Foundational Models (§2.2.1)

(2) Basic Models (§2.2.2)

(3) Terrain Models (§2.2.3)

(4) Supplementary Models (§2.2.4)

(5) Stochastic Fading Models (§2.2.5)

(6) Many-Ray (Ray-Tracing) Models (§2.2.6)

Name	Short Name	Category	Coverage Notes	Citations	Year
Friis' Free-space	friis	Foundational	$d > 2a^2/\lambda$	[77]	1946
Egli	egli	Basic	$30MHz < f < 3GHz$	[69, 206]	1957
Hata-Okumura	hata	Basic	$1km < d < 10km$; $150 <= f <= 1500MHz$ $30 <= h_1 <= 200m$; $1 <= h_1 <= 20$	[157]	1968
Edwards-Durkin	edwards	Basic/Terrain		[68, 62]	1969
Allsebrook-Parsons	allsebrook	Basic/Terrain	$f \in 85, 167, 441MHz$; Urban	[21, 62]	1977
Blomquist-Ladell	blomquist	Basic/Terrain		[37, 62]	1977
Longley-Rice Irregular Terrain Model (ITM)	itm	Terrain	$1km < d < 2000km$ $20MHz < f < 20GHz$	[98, 99]	1982
Walfisch-Bertoni	bertoni	Basic		[235]	1988
Flat-Edge	flatedge	Basic		[203]	1991
TM90	tm90	Basic	$d <= 10miles; h_1 <= 300feet$	[58]	1991
COST-231	cost231	Basic	$1km < d < 20km$;	[48]	1993
Walfisch-Ikegami	walfish	Basic	$200m < d < 5km$; $800MHz < f < 2GHz$; $4m < h_b < 50m$; $1m < h_m < 3m$	[48, 153, 34]	1993
Two-Ray (Ground Reflection)	two.ray	Foundational		[183, 206, 165]	1994
Hata-Davidson	davidson	Basic	$1km < d < 300km$; $150MHz < f < 1.5GHz$; $30m < h_b < 1500m$; $1m < h_m < 20m$	[38, 153]	1997
Oda	oda	Basic		[154]	1997
Erceg-Greenstein	erceg	Basic	$f \approx 1.9GHz$; Suburban	[71]	1998
Directional Gain Reduction Factor (GRF)	grf	Supplementary	Dir. Recv. Ant.,$f \approx 1.9GHz$	[85]	1999
Rural Hata	rural.hata	Basic	$f \in 160, 450, 900MHz$; Rural (Lithuania)	[143]	2000
ITU Terrain	itu	Terrain		[206, 107]	2001
Stanford University Interium (SUI)	sui	Basic	$2.5 < f < 2.7GHz$	[72, 19]	2001
Green-Obaidat	green	Basic		[84]	2002
ITU-R	itur	Basic	$1km < d < 10km$; $1.5GHz < f < 2GHz$; $30m < h_b < 200m$; $1m < h_m < 10m$	[107, 153]	2002
ECC-33	ecc33	Basic	$1km < d < 10km$; $700 <= f <= 3000MHz$ $20 <= h_1 <= 200m$; $5 <= h_1 <= 10$	[66, 19]	2003
Riback-Medbo	fc	Supplementary	$460MHz < f < 5.1GHz$	[190]	2006
ITU-R 452	itur452	Terrain		[109]	2007
IMT-2000	imt2000	Basic	Urban	[78]	2007
deSouza	desouza	Basic	$f \approx 2.4GHz; d < 120\,m$	[61]	2008
Effective Directivity Antenna Model (EDAM)	edam	Supplementary	Directional Antennas; $f \approx 2.4GHz$	[28]	2009
Herring Air-to-Ground	herring.atg	Basic	$f \approx 2.4GHz$	[94]	2010
Herring Ground-to-Ground	herring.gtg	Basic	$f \approx 2.4GHz$	[94]	2010

Table 2.1: *A priori* models studied along with their categorization, required input, coverage remarks, relevant citations, and year of (initial) publication.

Each category and its respective subcategories are discussed in turn in the following subsections. Table 2.1 provides a chronological list of the models discussed here and provides their major category, coverage, and initial publication.

2.2.1 Theoretical/Foundational Models

The first models worth considering are purely analytical models derived from the theory of idealized electromagnetic propagation. Although these models are questionably accurate, they are simple to understand and implement. As a result they have been widely adopted into network simulators and other applications, and often serve to compute a minimum loss for other, more complex, models.

2.2.1.1 Free-space Between Isotropic Antennas

In [77], Friis proposed a basic formula for free-space transmission loss:

$$\frac{P_{rx}}{P_{tx}} = \frac{A_{rx}A_{tx}}{d^2\lambda^2} \tag{2.5}$$

This formula describes the ratio between received power (P_{rx}) and transmitted power (P_{tx}) in terms of the effective area of the transmitting antenna (A_{tx}), receiving antenna (A_{rx}), the distance between (d) in meters, and the wavelength of the carrier (λ) in meters. For ideal isotropic antennas, this formula can be simplified to:

$$\frac{P_{rx}}{P_{tx}} = \left(\frac{\lambda}{4\pi d}\right)^2 \tag{2.6}$$

Or, more commonly, we solve for the power at the receiver in terms of the power from the transmitter and the path loss:

$$P_{rx} = P_{tx}\left(\frac{\lambda}{4\pi d}\right)^2 \tag{2.7}$$

Converting equation 2.7 to take distance in km instead of m, frequency in MHz instead of wavelength in m, and converting the linear domain power units (W) to log domain units (dBm), gives the commonly

d	distance between transmitter and receiver along line of sight path in km
d_m	distance between transmitter and receiver along line of sight path in m ($1000d$)
h_{tx}/h_{rx}	height of transmitter/receiver above ground in m
P_{tx}/P_{rx}	power produce by transmitter/received by receiver
f	carrier frequency in MHz
λ	carrier wavelength in m
g_{tx}/g_{rx}	gain of the transmitters/receiver's antenna in the azimuthal direction of the transmitter
θ	angle from transmitter to receiver in azimuthal plane relative to true north
θ'	angle from receiver to transmitter ...
ϕ	angle of elevation between transmitter and receiver relative to horizontal (inclination)
ϕ'	angle from receiver to transmitter ...
$U(a,b)$	a uniformly distributed random variable between a and b (inclusive)
$N(\mu,\sigma)$	a normally distributed random variable of mean μ and standard deviation σ
$LN(\mu,\sigma)$	a lognormally distributed random variable of mean μ and standard deviation σ
R	the radius of the earth in m ($\approx 6.371 * 10^6$)
C	the speed of light in m/s ($\approx 299.792 * 10^6$)
ϵ_r	relative permittivity (of obstructing material)

Table 2.2: Commonly used mathematical symbols.

seen reference equation for path loss as a function of carrier frequency and distance:

$$P_{rx} = P_{tx} - (20log_{10}(d) + 20log_{10}(f) + 32.45) \tag{2.8}$$

Where power in decibels relative to a milliwatt (dBm) can be obtained from power in Watts (W) using this conversion:

$$P_{dBm} = 10log_{10}(P_{mW}) \tag{2.9}$$

2.2.1.2 Flexible Path Loss Exponent

Whereas Friis' equation assumes that signal degrades as a function of d^2, a common extension to No(n) Line of Sight (NLOS) environments is to use a larger exponent. To allow for this, we simply substitute in α, which can be set to any value greater than zero, but is most commonly set to 2:

$$P_{rx} = P_{tx} - (10\alpha log_{10}(d) + 20log_{10}(f) + 32.45) \tag{2.10}$$

Often, this model will be given relative to some reference distance d_0 (commonly 100m), where the assumption is that several measurements are made at this distance, and those values are used to fit a slope:

$$P_{rx} = P_{tx} - (10\alpha log_{10}(d/d_0) + 20log_{10}(f) + 32.45) \tag{2.11}$$

2.2.1.3 Ground Reflection

As a modest extension to the free-space path loss model, the Two-Ray Ground Reflection model considers a second path that reflects from the ground between the transmitter and receiver [183, 206, 165]. First, we calculate the break distance:

$$d_c = (4\pi h_{tx} h_{rx})/\lambda \tag{2.12}$$

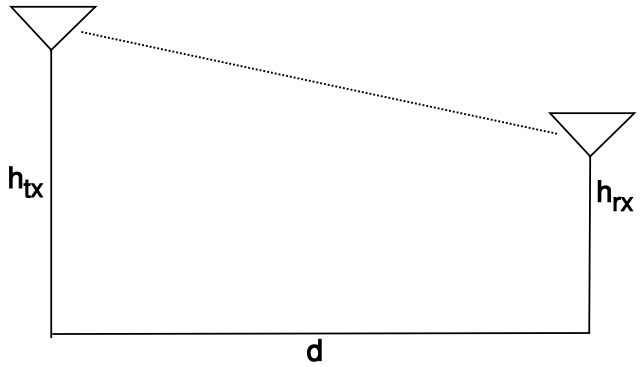


Figure 2.4: Schematic of link geometry used by basic models.

where h_{tx} and h_{rx} are the heights of the transmitter and receiver antennas, respectively (in m). For distances shorter than this break distance, we simply use Friis' equation as the receiver is not far enough away to receive a substantial ground reflected path loss. For distances longer than the break distance, we use the modified path loss formula:

$$P_r = \frac{P_{tx} h_{tx}^2 h_{rx}^2}{d^4} \tag{2.13}$$

In [154], Oda *et al.* propose a minor extension to this model where the plane of reflection is considered to be above the nominal ground clutter, and a factor for probability of collision per unit distance is considered. By adjusting this height offset (h_0), the reflectivity coefficient (R) and this negative exponentially weighted factor, one can coax the simple Two-Ray model into more closely fitting some types of measured data.

2.2.2 Basic Models

Basic models are the most numerous of the model types. They compute path loss along a single path and often use corrections based on measurements made in one or more environments. In general, they use the distance, carrier frequency, and transmitter and receiver heights as input. Figure 2.4 gives a schematic of the basic link geometry. Some models also have their own esoteric parameters to select between different modes of computation or fine tuning. Here we subdivide these models into deterministic and stochastic. The stochastic models use one or more random variables to account for channel variation (and hence, can predict

a distribution instead of a median value).

2.2.2.1 Egli

The Egli Model [69] is an early empirical model based on measurements made in New York City and parts of New Jersey by John Egli. The simplified version, based on extracting a model from numerous graphs and nomograms [62] and [34] is:

$$P_{rx} = P_{tx} - 20log_{10}(f) + 40log_{10}(d) - 20log_{10}(h_{tx}) + k \tag{2.14}$$

with

$$k = \begin{cases} 76.3 - 10log_{10}(h_{rx}) & h_{rx} \leq 10 \\ 85.9 - 20log_{10}(h_{rx}) & h_{rx} > 10 \end{cases} \tag{2.15}$$

2.2.2.2 Green-Obaidat

The Green-Obaidat model suggested in [84] is a small modification to free-space path loss that adjusts for the relative heights of the transmitter and receiver and uses a path loss exponent of $\alpha = 4$:

$$P_{rx} = P_{tx} - (40log_{10}(d) + 20log_{10}(f) - 20log_{10}(h_{tx}h_{rx})) \tag{2.16}$$

In this model, d is given in km, h_{tx} and h_{rx} in m, and f in MHz. The authors perform minimal validation using 802.11 devices operating at 2.4 GHz. This model is one of several that extends basic models to include the relative height of nodes in their calculations—in this case, the heights are multiplied.

2.2.2.3 Edwards-Durkin

The Edwards-Durkin model [68] simply sums classical free-space loss (lf) with an additional correction due to plane earth propagation loss from Bullington [42]:

$$lp = 118.7 - 20log10(h_{rx}) - \tag{2.17}$$

$$20log10(h_{tx}) + 40log10(d)$$

$$PL = lf + lp \tag{2.18}$$

The constants in this formula are fitted from empirical measurements made in the United Kingdom by Durkin [67]. In [62], Delisle updates this model with a statistical terrain diffraction loss estimate ($ld(\Delta h)$, described in section 2.2.4.2) and leaving out the free-space term:

$$PL' = lp + ld(\Delta h) \tag{2.19}$$

2.2.2.4 Blomquist-Ladell

The Blomquist-Ladell model [37] is similar in construction to the Edwards-Durkin model. It computes an excess plane earth loss, with a correction factor, and sums it with classical free-space loss. As with the Edwards-Durkin model, it can be extended with a statistical terrain diffraction loss estimate. The fitted constants in this model were derived from measurements in the VHF and UHF bands over rolling terrian in Sweden.

The excess plane earth loss is computed as:

$$fb = 10.0log10\,|a_{tx}a_{rx}| + y \tag{2.20}$$

$$a_i = \frac{4\pi h_i^2}{\lambda d_m} + \frac{\lambda \epsilon_r^2}{\pi d(\epsilon_r - 1)} \tag{2.21}$$

With d being the link distance in meters, and the correction factor, y:

$$y = \begin{cases} -2.8x & x < 0.53 \\ 6.7 + 10log10(x) - 10.2x & \text{o.w.} \end{cases} \tag{2.22}$$

$$x = (2\pi/\lambda)^{1/3}(kR)^{-2/3}d \tag{2.23}$$

Finally, the path loss is summed with free-space loss:

$$s_+ = lf + \sqrt{fb^2 + ld(\Delta h)^2} \tag{2.24}$$

$$s_- = lf - \sqrt{fb^2 - ld(\Delta h)^2} \tag{2.25}$$

$$PL = \begin{cases} s_+ & fb \leq 0 \\ s_+ & fb > 0, \leq |ld(\Delta h)| \\ s_- & fb > 0, > |ld(\Delta h)| \end{cases} \tag{2.26}$$

(2.27)

Where $ld(\Delta h)$ is the statistical terrain diffraction loss estimate described below in section 2.2.4.2, lf is the basic free-space loss calculated as in equation 2.8, k is the earth radius factor (typically 4/3), and ϵ_r is the dielectric constant (relative permittivity) of the ground (Delisle recommends 10 for dry earth).

2.2.2.5 Allsebrook-Parsons

The Allsebrook-Parsons model [21] is an extension to the Blomquist-Laddell model that adds an additional loss due to buildings. The authors based the empirical adjustment on measurements taken in British cities. The model also suggests a constant additional loss (named γ here) of 13 dB for frequencies above 200 MHz[1] .

$$lb = 20log10\left(\frac{h_0 - h_{rx}}{548\sqrt{d_2 10^{-3} f}}\right) + 16 \tag{2.28}$$

$$PL = PL_b + lb + \gamma \tag{2.29}$$

where PL_b is the path loss computed by the Blomquist-Ladell model, h_0 is the average height of buildings in the neighborhood of the mobile station in m, and d_2 is the average street width in m.

[1] The validity of this correction is questioned in [62].

2.2.2.6 deSouza-Lins

In [61], de Souza and Lins present an entirely empirical model explicitly fitted to data collected at 2.4 GHz. This model is a function of distance (in meters) and relative humidity percentage (h):

$$P_{rx} = P_{tx} - (\beta_0 + \beta_1 log_{10}(d) + \beta_2 d + \beta_3 log_{10}(h)) \tag{2.30}$$

Although the authors claim very impressive performance at the sites (two indoor, two outdoor) they study (from which the fitted β parameters are derived), the short distances studied ($< 120\,m$) suggest that this model may be inappropriate for modeling lengthier links.

2.2.2.7 TM90

In [58], the authors propose a propagation model intended for suburban areas and for propagation distances less than 10 miles. This model is very simple, using a flexible path loss exponent model with $\alpha = 4$, accounting for antenna heights as in the Hata-Okumura model, and adding an additional loss for average building penetration (outdoor-indoor interface loss). This model is the FCC recommended model for shorter propagation distances (as opposed to the Irregular Terrain Model (ITM), which is recommended by the FCC for long links).

2.2.2.8 Hata-Okumura

The Hata-Okumura model is an empirical model where measurements made by Okumura in and around Tokyo, Japan are approximated with equations proposed by Hata [34, 157]. The model is considered valid for frequencies from 150 MHz to 1500 MHz, transmitter heights between 30 m and 200 m, receiver heights between 1 m and 10 m, and distances greater than 1 km. The model takes an additional environment parameter that can be one of “open”, “suburban”, “urban medium”, or “urban large”, which selects among different modes of computation for differing levels of environment complexity (as related to population density).

The correction factors are first computed, based on the enviromnental complexity:

$$a = \begin{cases} 3.2log10((11.75h_{rx})^2) - & \text{large city} \\ 4.97 & \\ h_{rx}(1.1log10(f) - 0.7) - & \text{o.w.} \\ (1.56log10(f) - 0.8) & \end{cases} \tag{2.31}$$

$$k = \begin{cases} 2log10(f/28)^2 + 5.4 & \text{suburban} \\ 4.78log10(f)^2 - 18.33 & \text{open} \\ log10(f) + 40.94 & \\ 0 & \text{o.w.} \end{cases} \tag{2.32}$$

(2.33)

Then, the final path loss is computed by offsetting a constant (transmitter height adjusted) free-space path loss (b):

$$b = 69.55 + 26.16log10(f) - 13.82log10(h_{tx}) \tag{2.34}$$

$$PL = b - a + log10(d) * (44.9 - 6.55log10(h_{tx})) - k \tag{2.35}$$

Due to the popularity of the Hata-Okumura model, there have been numerous extensions and corrections:

COST-Hata/Extended Hata

The COST-Hata model is an extension of the Hata-Okumura model to cover frequencies up to 2000 MHz. It was proposed as part of the COST-231 [34, 78, 48].

First the correction factors a and c are computed:

$$a = h_{rx}(1.1log10(f) - 0.7) - \tag{2.36}$$
$$(1.56log10(f) - 0.8)$$

$$c = \begin{cases} 3.0 & \text{large city} \\ 0.0 & \text{o.w.} \end{cases} \tag{2.37}$$

$$\tag{2.38}$$

Then, as before the path loss is computed by offsetting a free-space path loss computation (b):

$$b = 46.33 + 33.9log10(f) - \tag{2.39}$$
$$13.82log10(h_{rx})$$

$$PL = b - a + (44.9 - 6.55log10(h_{rx})) * \tag{2.40}$$
$$log10(d) + c$$

Hata-Davidson

In the Telecommunications Industry Association (TIA) recommendation TSB-88-B, an extension to the Hata-Okumura model is proposed, which provides corrections for long links (up to 300 km) and high transmitters (up to 2500 m) [223, 153]:

$$a = \begin{cases} 0.62137(d - 20.0)* & d \geq 20km \\ (0.5 + 0.15log10(\frac{h_{tx}}{121.92})) & \\ 0 & \text{o.w.} \end{cases} \tag{2.41}$$

$$s_1 = \begin{cases} 0.174(d - 64.38) & d \geq 64.38km \\ 0 & \text{o.w.} \end{cases} \tag{2.42}$$

$$s_2 = \begin{cases} |0.00784log10(9.98/d)| & h_{tx} > 300m \\ (h_{tx} - 300.0) & \\ 0 & \text{o.w.} \end{cases} \tag{2.43}$$

$$s_3 = (f/250)log10(1500/f) \tag{2.44}$$

$$s_4 = \begin{cases} 0.112log10(1500/f)(d - 64.38) & d > 64.38 \\ 0 & \text{o.w.} \end{cases} \tag{2.45}$$

$$PL_{davidson} = PL_{hata} + (a - s1 - s2 - s3 - s4) \tag{2.46}$$

Where a, s_1, s_2, s_3, and s_4 are used to correct the calculation in equation 2.36.

ECC-33

In [66], the Electronic Communication Committee (ECC) within the European Conference of Postal and Telecommunications Administrations (CEPT) extend the coverage up to 3,500 MHz:

$$afs = 92.4 + 20.0log10(d) + 20.0log10(f) \tag{2.47}$$

$$abm = 20.41 + 9.83log10(d) + 7.894log10(f) + 9.56log10(f)^2 \tag{2.48}$$

$$gb = log10(hb/200)(13.958 + 5.8log10(d)^2) \tag{2.49}$$

$$gr = \begin{cases} (42.57 + 13.7 * log10(f))(log10(h_{rx}) - 0.585) & \text{medium city} \\ 0.0 & \text{o.w.} \end{cases} \tag{2.50}$$

$$PL_{ecc33} = afs + abm - gb - gr \tag{2.51}$$

ITU-R/CCIR

The International Radio Consultive Committee (CCIR) (now the International Telecommunications Union Radiocommunication Sector (ITU-R)) proposed a version of the Hata-Okumura model, which takes a real-valued parameter, the percentage of area covered by buildings (bp), instead of a discrete environment class. This model is an attempt at correcting systematic underestimations observed in the Hata-Okumura model and is in essence the Hata-Okumura model for "urban-medium" environments with an additional correction factor related to the new parameter [153]:

$$a = (1.1log10(f) - 0.7)h_{rx} - (1.56log10(f) - 0.8) \tag{2.52}$$

$$b = \begin{cases} 30 - 25log10(bp) & bp > 0 \\ 0 & \text{o.w} \end{cases} \tag{2.53}$$

$$c = 69.55 + 26.16log10(f) - 13.82log10(h_{tx}) \tag{2.54}$$

$$PL_{ccir} = c - a + (44.9 - 6.55log10(h_{tx}))log10(d) - b \tag{2.55}$$

Rural Hata

In [143], Medeisis *et al.* propose a correction for the classic Hata-Okumura model as defined in ITU-R 529 to correct for systematic overestimations of path loss in rural terrain. Their model proposes new fitted values for the path loss exponent and fixed offset to replace those that are defined in the default model. These fits are obtained from data collected using a simple random sampling scheme in rural Lithuania at three frequencies below 900 MHz. In addition to this, the authors propose a method to do site-specific fitting in a similar way so that their approach can be used in other environments and at other frequencies.

The loss in dBuV/m (decibels relative to a microVolt per meter) is given by the equation:

$$e_{sys} = -6.16log(f) + 13.82 * log(h_{tx}) + ((1.1log(f) - 0.7)h_{rx} - (1.56log(f) - 0.8)) \tag{2.56}$$

$$\gamma_{sys} = \gamma(44.9 - 6.55log(h_{tx})) \tag{2.57}$$

$$PL'_{rural} = e_0 + e_{sys} + \gamma_{sys}log(d) \tag{2.58}$$

To convert this value to dBm so it is consistent with our other equations, we use the following conversion [18]:

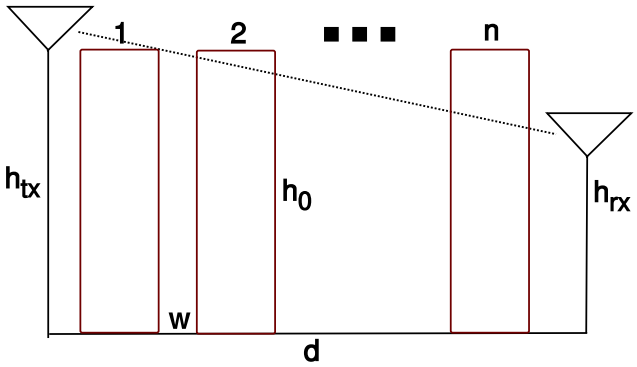


Figure 2.5: Schematic of link geometry used by the Flat-Edge family of basic models.

$$PL_{rural} = -(PL'_{rural} + g_{rx} - 20log(f) - 77) \tag{2.59}$$

2.2.2.9 Flat-Edge

The Flat-Edge model, proposed by Saunders and Bonar [203] takes a very different approach as compared to the Hata family of models. Saunders and Bonar propose a model that computes approximate knife-edge diffraction losses due to multiple obstructions (buildings) that are regularly spaced. Figure 2.5 provides a schematic of this setup. The model takes as parameters the number of obstructions between the transmitter and receiver (n), the constant distance between them (w), and their constant height (h_0). The assumption is that there is a transmitter either above or below a series of obstacles of the constant size and spacing and the receiver is below the top of the buildings. The model works by summing the loss due to Fresnel obstruction by the obstacles, the basic free-space propagation loss (lf), computed using equation 2.8, and the loss due to diffraction over the final obstruction.

First, the value of t is calculated:

$$t = \phi\sqrt{\frac{\pi w}{\lambda}} \tag{2.60}$$

If $1 \leq n \leq 100$ and $-1 \leq t < 0$, then the approximate fit due to Barclay [34] is used:

$$ln = -(3.29 + 9.90log10(t) - (0.77 + 0.26log10(n))) \tag{2.61}$$

If, however, this is not the case, then a complicated series of Fresnel calculations are required to compute ln. Those equations are well summarized in [203] Appendix B.3. The additional loss due to diffraction over the final obstruction is calculated using the method of Ikegami [100].

$$le = \begin{cases} 10log10(f) + 10log10(sin(\phi)) + 20log10(h_0 - h_{rx}) - \\ 10log10(w) - 10log10(1 + 3/lr^2) - 5.8 & h_{rx} < h_0, \phi \neq 0 \\ 0 & \text{o.w.} \end{cases} \tag{2.62}$$

where lr is the refraction loss fraction, commonly set to 0.25. Finally, the sum is computed:

$$PL_{flatedge} = ln + lf + le \tag{2.63}$$

2.2.2.10 Walfisch-Bertoni

The Walfisch-Bertoni model is the limiting case of the Flat-Edge model when the number of buildings is large enough for the field to settle [34]. Hence, this model takes as parameters the distance between obstructions and their nominal size, but not the number of them, which is implicit to the calculation.

$$la = \begin{cases} log10((w/2) + (h_0 - h_{rx})^2) - 9log10(w) + \\ 20log10(atan((2.0(h_0 - h_{rx}))/w)) & h_{rx} \leq h_0 \\ 0 & \text{o.w} \end{cases} \tag{2.64}$$

$$c = \begin{cases} 18log10(h_{tx} - h_0) & h_{tx} - h_0 > 0 \\ 0 & \text{o.w} \end{cases} \tag{2.65}$$

$$lex = 57.1 + log10(f) + 18log10(d) - c - 18log10(1 - d^2/(17(h_{rx} - h_0))) \tag{2.66}$$

$$PL_{bertoni} = lf + lex + la \tag{2.67}$$

where lf is the trivial free-space loss computed with equation 2.8.

2.2.2.11 Walfisch-Ikegami

The European Cooperation in the field of Scientific and Technical Research Action 231 (COST-231)/Walfisch-Ikegami is a compromise proposal by the COST-231 that combines the Walfisch-Bertoni model with an additional reflection down to the receiver using the Ikegami model [100] along with some empirical corrections from measurements [34]. The model distinguishes between Line of Sight (LOS) propagation and NLOS and uses different calculations for each. In addition to the expected parameters describing the geometry of the LOS path, this model requires specification of the constant building height, street width (w), distance between buildings (b, such that $b - w$ is the nominal building width), the angle of the incident wave to the street (π radians for vertically polarized antennas, 0 for horizontal), and the building size (either "medium" or "large").

For NLOS links, the model includes calculations for excess loss above free-space loss due to roof-to-street diffraction loss and multiscreen diffraction loss. After calculating this excess loss, if it is positive, it is summed with the free-space loss and used. Otherwise, uncorrected free-space loss is returned. For LOS links, the returned value is free-space loss with a fudge factor to attempt to avoid underestimates: $6 * log_{10}(50 * d)$. Because this model is reasonably complicated, we refer the interested reader to the excellent slides maintained by the National Institute of Standards and Technology (NIST) [153] for further details.

The remaining basic models include a random variate (stochastic) term that attempts to capture the time-varying nature of the wireless channel due to small scale fading.

2.2.2.12 Herring

The Herring model is a recent proposal by Herring *et al.* [94]. The model proposes two distinct models, one for Air-to-Ground (ATG) communications and one for Ground-to-Ground (GTG), both of which are based on fits to data collected by the authors at 2.4 GHz in Cambridge, Massachusetts. The ATG model is a simple error term on top of the free-space path loss model:

$$P_{rx} = P_{tx} - (lf + N(30, 8.3)) \tag{2.68}$$

where lf is calculated as in equation 2.8 and $N(30, 8.3)$ is a random Gaussian with mean 30 and standard deviation of 8.3. The GTG model is slightly more complex. It first computes a random Gaussian path loss exponent with uniform random offset:

$$\alpha = U(2, 5) + N(0, 0.22) \tag{2.69}$$

This path loss exponent is then used along with a larger excess loss value:

$$P_{rx} = P_{tx} - (lf(\alpha) + N(40, 5.5)) \tag{2.70}$$

where $lf(\alpha)$ is computed as in equation 2.10.

2.2.2.13 Erceg-Greenstein

In [71], Erceg *et al.* present a measurement-based model for path loss around 1.9 GHz using a large data set collected by AT&T in suburban locations around New Jersey. It is a fitted model that combines a fit for median path loss at some distance d and a randomly distributed variation:

$$\begin{aligned} PL = A + 10(a - b * h_{tx} + \\ (\frac{c}{h_{rx}})log10(\frac{d}{d_0}) + x10log10(\frac{d}{d_0}) + \\ y\mu_\sigma + yz\sigma_\sigma \end{aligned} \tag{2.71}$$

where the values of $a, b, c, \sigma_\gamma, \mu_\sigma$, and σ_σ are fitted parameters for each of the three terrain categories: hilly with moderate to heavy tree density (A), hilly with light tree density or flat with moderate to heavy tree density (B), or flat with light tree density (C). The value A is the trivial free-space path loss (from equation 2.8, for instance) at some reference distance (d_0, usually 100 m). And, x, y, and z are normally distributed random variables between -2 and 2 (x is between -1.5 and 1.5).

2.2.2.14 IMT-2000: Pedestrian Environment

Three path loss models for IMT-2000/3G are provided in [78], one for the indoor office environment, one for the outdoor to indoor and pedestrian environment, and one for the vehicular environment. It is the pedestrian model which we describe here, which is simply equation 2.10 with $\alpha = 4$, a constant (optional) offset for building penetration loss (k_1) and a lognormally distributed offset to account for shadowing loss (k_2):

$$P_{rx} = P_{tx} - (40log_{10}(d) + 30log_{10}(f) + k_1 + k_2 + 21) \tag{2.72}$$

with

$$k_1 = \begin{cases} 18 & indoors \\ 0 & o.w. \end{cases} \tag{2.73}$$

and

$$k_2 = LN(0, 10) = e^{0+10N(0,1)} \tag{2.74}$$

where $LN(0, 10)$ is a lognormally distributed random variable with zero mean and a standard deviation of 10.

2.2.3 Terrain Models

Terrain models are similar to the basic models, but also attempt to compute diffraction losses along the line-of-sight path due to obstructions (terrain or buildings, for instance, see figure 2.6 for a schematic). They are an order of magnitude more complex, but are immensely popular especially for long propagation distances at high power in the VHF band (i.e., television transmitters). Because of the relative complexity of these models, the reader will need to refer to the citations for details of the implementations. Here, their functionality is summarized at a high level.

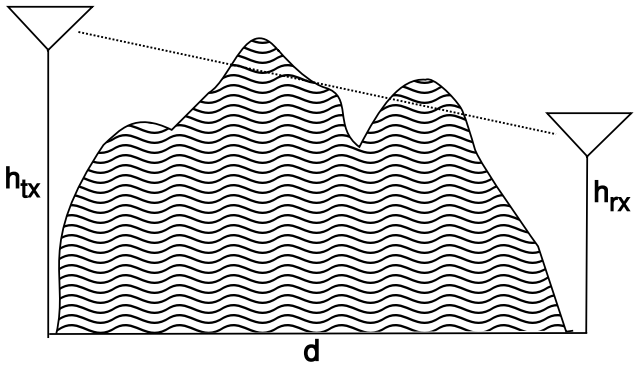


Figure 2.6: Schematic of link geometry used by terrain models.

2.2.3.1 ITU Terrain

The ITU terrain model is a simple model that combines free-space path loss with a single diffraction due to terrain [206, 107]. A Digital Elevation Model (DEM) is used to compute the loss due to the most significant path obstruction in terms of Fresnel zone blockage. In the event that the transmission path has no blockage, then free-space path loss (with an exponent of 2) is used. The radius of the first Fresnel zone is computed at the site of blockage:

$$f_1 = 17.3\sqrt{\frac{d_1 d_2}{fd}} \tag{2.75}$$

where d_1 is the distance from the transmitter to the obstruction, d_2 is the distance from the receiver to the obstruction, d is the total distance, and f is the carrier frequency. The constant, 17.3, is derived from equations governing the physical optics of Fresnel lenses. Then the additional path loss (outside of free-space) is used for this blockage:

$$al = -20.0h/f_1 + 10.0 \tag{2.76}$$

The model suggests that a negative loss due to the blockage (which is actually a gain, i.e., negative loss) or any loss less than 6 dB should be discarded. The maximum additional loss is then used to “correct” the free-space loss assumption.

2.2.3.2 ITU-R 452

The clear-air interference prediction algorithm described in ITU-R 452 [109] serves a fine example of the state of the art in terrain path loss models. This model makes a prediction of median path loss based on the sum of free-space path loss with several corrections:

(1) losses from knife-edge diffractions over terrain obstacles

(2) losses from absorption due to atmospheric gases (water vapor)

(3) losses from tropospheric scatter, ducting, coupling, and layer reflection in the atmosphere

(4) losses due to obstruction from the curvature of the earth

(5) additional clutter losses derived from land cover classification [106] near the transmitter and receiver

The model computes the path loss in terms of a confidence value p, which is the not-to-exceed probability. Using $p = 50$ computes a median value, $p = 100$ computes a worst-case value and $p = 0$ computes a best-case value.

In addition to p, the model takes a handful of other parameters: Δn, which is the radio refractivity of the earth between the transmitter and receiver (values between 35 and 60 are typical for the environments studied in this thesis), n_0, which is the surface level refractivity, and ω, which is the fraction of the path over water (i.e., for intercontinental links. $\omega = 0$ for all of our environments). This model is leaps and bounds more complex than those presented above, requiring a tremendous number of calculations often based on numerical approximations (i.e., knife-edge diffraction).

ITU-R 452 suggests additional extensions for modeling the interference due to hydrometers such as rain and weather cells. This adds substantial complexity to the algorithm with negligible benefit for many communications applications operating in the upper end of the VHF band. Indeed, many of the parameters computed by the ITU-452 model are negligible for commonly used communications frequencies (for instance, absorption due to atmospheric gases). In [237], Whitteker suggests a similar model which shares many of the attributes of ITU-R 452, with slightly less complexity.

2.2.3.3 Longley-Rice Irregular Terrain Model

The ITM [98, 99] may be the most widely known general purpose path loss model and is used in a number of popular network planning tools (e.g., [136, 54]). This model was developed by the NTIA in order to plan deployments of Very High Frequency (VHF) transmitters (i.e., broadcast television in the US). Hence, much like the ITU-R model, it is designed for very long transmission at high power from well-positioned transmitters. For this reason its applicability to modeling propagation in, e.g., urban microcells, is questionable at best. Much like ITU-R 452, the ITM computes excess loss from free-space by considering knife-edge diffractions at terrain obstacles, losses due to the curvature of the earth, and tropospheric scatter. The principle difference is that ITU-R 452 includes some calculation for local clutter losses based on land cover classification data, otherwise the models can be thought to be quite similar.

2.2.4 Supplementary Models

The next category of models are supplementary models, which cannot stand on their own, but are instead used to make corrections to existing (complete) models. Here we subdivide the models by the phenomenon they wish to correct for.

2.2.4.1 Frequency Coverage

The Riback-Medbo model [190] attempts to correct for the (ill) effects of using a model intended for one frequency at a different frequency. The algorithm the authors propose provides a fitted correction when given the computed path loss, the assumed frequency, the target frequency based on measurements they make a three different frequencies:

$$a = 0.09 \tag{2.77}$$

$$b = 256 * 10^6 \tag{2.78}$$

$$c = 1.8 \tag{2.79}$$

$$k = a(atan(f_0/b - c) - atan(f_0/b - c)) \tag{2.80}$$

$$PL_{fc} = PL_0 + 20log10(f/f_0) - k(PL_0 - A) \tag{2.81}$$

where k is the correction factor which is used to correct the path loss value (PL_0) at a given frequency (f_0) so that it is better able to predict the loss at the desired frequency (f). The value A is the trivial free-space loss (from equation 2.8) at the original frequency (f_0). The authors validate this model using a significant amount of data in a limited number of (suburban) environments, from which the empirical constants are derived.

2.2.4.2 Obstructions

Obstruction models account for losses due to specific obstructions along the main (or some secondary) path. They are the most numerous and varying of the supplementary models:

Atmospheric Gases

The effects due to absorption by atmospheric gases are minimal at UHF frequencies and totally negligible at higher frequencies. However, it is worth noting that such corrective models are available for water vapor and to a lesser extent for other gases (e.g., [110]).

Statistical Terrain Diffraction Estimate

Because terrain information is not always available and computing individual diffractions over terrain can be computationally costly, [62] proposes a method for computing an estimate of additional losses due to terrain. In addition to the geometry of the line-of-sight path, this approach makes use of a single parameter, Δh, which describes the "roughness" of the terrain. A value of ≈ 15 is considered minimal, ≈ 200 is used for hilly terrain, and ≈ 400 for very rugged terrain. In [62], Delisle *et al.* propose the use of this estimate in combination with other models, such as Allsebrook-Parsons, Blomquist-Ladell, and Edwards-Durkin. In

this way, it can be used to retrofit any basic model with corrections for losses from terrain obstacles and clutter.

Building-Transmission

The Building-Transmission model proposed by de Jong *et al.* in [59] attempts to model the loss due to transmission *through* a building in an urban environment. The authors attempt to isolate this effect from fades along other paths and instead present a statistical model for just the loss encountered by transmission through a number of representative buildings at 1.9 GHz. They find that on average there is a loss of approximately 2.1 dB/m at this frequency and use this to develop an algorithm to compute total transmission loss, including refraction at the exterior walls. For this model to be of use in practice, one must know the positions and shape of buildings along with the permittivity and conductivity of the buildings' outer surfaces.

Durgin-Rapaport

In [64], Durgin *et al.* make numerous measurements around residential homes and trees at 5.85 GHz. They use the collected data to come up with constant fitted values for losses associated with outdoor-indoor interface loss, loss due to single trees and stands of trees, as well as interior walls. These values are then used to form the basis of a "partition" path loss model that computes the final signal strength by computing the free-space loss and then combining it with the summed loss associated with each obstruction. A model of the same flavor and by the same authors is also proposed in [135], but for 2.4 GHz.

Vegetation

There have been a number of works that attempt to, in one way or another, model the losses due to vegetation obstructions. [108] proposes a very complex formulation that attempts to model the diffraction above and around a stand of trees. Parameters are provided for several species of trees, both in leaf and out of leaf. In [224], Torico *et al.* present an interesting but largely impractical theoretical model for loss due to trees. In this work, trees are modeled as a screen containing randomly placed cylindrical defractors. Although not useful for general prediction, this model demonstrates that vegetation can cause substantial losses. In [45], Chee *et al.* present a similar analytical model. The lack of availability of vector data describing the location, shape, and type of vegetation prohibits use in most applications. A more practical proposal is described in [134], where rain forest vegetation is modeled using four layers (ground, trees,

foilage, sky) with different propagation characteristics and interlayer ducting.

2.2.4.3 Directivity

Directivity models attempt to account for multipath (scattering) losses that are unique to situations where the transmitter, or more importantly the receiver, is using a directional antenna. The problem here is that directional antennas "emphasize" some azimuthal directions more than others, which leads to nontrivial multipath effects at the receiver. If the goal is to model a link involving directional antennas and the antenna is assumed isotropic (perhaps with the gain assumed to be equal to the maximum gain of the main lobe), a substantial deviation from reality can occur.

Gain Reduction Factor

In [85], Greenstein and Erceg find that there can be substantial gain reduction at the receiver. The authors make measurements in suburban New Jersey at 1.9 Ghz and fit a model to the effects. The model is fitted to the beamwidth of the receiving antenna and whether or not the measurements are made in winter (i.e., with or without leaves on trees).

$$i = \begin{cases} 1 & winter \\ -1 & o.w. \end{cases} \tag{2.82}$$

$$\mu = -(0.53 + 0.1i)log(\beta/360) + (0.50 + 0.04i)log(\beta/360)^2 \tag{2.83}$$

$$\sigma = -(0.93 + 0.02i)log(\beta/360) \tag{2.84}$$

$$PL_{grf} = N(\mu, \sigma) \tag{2.85}$$

where PL_{grf} is the additional gain or attenuation in dB and β is the beamwidth of the receiving antenna in degrees.

EDAM

The EDAM is a bin-fitted model derived from a large number of measurements made in several representative environments (multiple indoor and multiple outdoor environments) both with commodity hardware and with special purpose hardware. The result is a model that, when given an environment class,

will provide a correction as a function of the gain pattern at the receiver in the direction of the transmitter and vice versa. The model is also able to be used in a stochastic fashion for a repeated measures approach and with or without a Gaussian-distributed fading correction. The model is described in detail in appendix A and in [28, 30], it was shown that this model is better suited to making path loss predictions in simulation-based evaluations involving directionality than standard models.

2.2.5 Stochastic Fading Models

Stochastic fading models add a random variable to a path loss model to account for additional fading in the wireless channel. This includes fades due to scattering and multipath effects that are uncorrelated in measurements over small distances (i.e., less than a wavelength). These fades are selective in both time and frequency, meaning that attentuation may vary as a function of either (or both). Stochastic fading models are especially useful in the design of physical layer/data-link layer of wireless networks.

A number of measurement studies, find that residual error in an explicit fit to measurements follows a lognormal distribution. This is equivalent to adding a zero mean normally distributed error term X_σ to equation 2.10:

$$P_{rx} = P_{tx} - (10\alpha log_{10}(d) + 20log_{10}(f) + 32.45 + X_\sigma) \tag{2.86}$$

This model is commonly referred to as the "lognormal shadowing" model and can be used as an empirically corrected model where values of α and σ are determined from measurements. This is the most coarse stochastic fading model and is usually considered to be appropriate only for modeling large scale effects [183].

Small scale (time varying) stochastic fading models typically look to either Rayleigh, Ricean, or Nakagami distributions. The inquisitive reader can refer to the excellent treatment by Skylar of Rayleigh and Ricean fading in [210] or [246, 148] for discussions of the Nakagami distribution. Some low-level applications may choose to explicitly model inter-symbol interference by determining the delay spread of arriving signals, as observed at the reciever, from a representative distribution. In [86], for instance, Greenstein et

al., show that both delay-spread and path gain appear to be lognormally distributed in their measurements at 900 MHz.

Barclay-Okumura

The Barclay-Okumura model is a simple model for stochastic fading proposed by Barclay in [34] based on data collected by Okumura. It can operate in either "urban" or "suburban" mode, and computes a zero-mean Gaussian distributed fade with standard deviation σ:

$$a = \begin{cases} 5.2 & urban \\ 6.6 & suburban \end{cases} \tag{2.87}$$

$$\sigma = 0.65 log_{10}(f)^2 - 1.3 log_{10}(f) + a \tag{2.88}$$

2.2.6 Many-Ray Models

Many-ray models are typically referred to as ray-tracing or ray-launching models in the literature. In the taxonomy proposed here, they are called "many-ray" models to highlight the way in which they differ from all of the aforementioned models: they attempt to calculate the path loss by summing the loss along many distinct paths instead of only the line-of-sight (LOS) path. These models require substantial and precise knowledge about the environment. Two-and-three dimensional vector models of buildings and interfering structures are the most commonly used data. These models trace the interaction of many individual paths and these obstacles, computing reflection, refraction, and diffraction using the UTD, or an equivalent numerical approximation. As a result, they are able to compute not only the median path loss predicted at the receiver, but also the delay spread (which can be used to compute Inter-symbol Interference (ISI)) and frequency shift (which can be used to model frequency-selective fading effects) of arriving signals.

Early papers in this area include the work of Ikegami *et al.* in [100] and Vogler in [231], where it is proposed that mean field strength be calculated by computing diffractions and reflections from building vector data. Some work has been done to increase the accuracy and speed of calculating diffractions (e.g., [203, 242] and the comparative discussion in [65]).

The early applications of these ideas were applied in two-dimensional ray-tracing models. In [248], Zhang and Moayeri propose a purely theoretical model that assumes a regular city grid and predicts a single reflected path (around corners) and a constant adjustment for other multipath effects. Different calculations are used based on whether the receiver is on a neighboring street or a side (perpendicular) street. In [202], Rustako *et al.* suggest that only 6 rays are necessary for modeling line-of-sight links in urban street-canyons. In [117], Kanatas *et al.* suggest a simple two-dimensional ray-tracing model that assumes a uniform rectilinear building/street layout and makes a minimal validation against measurements. In [198], Rizk *et al.* propose a two-dimensional ray-tracing approach that can deal with arbitrary building layouts and go to some effort to validate their approach. In [176], Piazzi *et al.* evaluate a two-dimensional ray-tracing approach in a residental environment and find decent results when the transmitter is positioned above the rooftops. In [80], the authors extend the Walfisch-Ikegami model to include corrections from ray-tracing and static adjustments for the presence of trees.

More recently, authors have proposed three-dimensional models that require substantially more computation. In [241], Wölfle *et al.* propose a three-dimensional ray-optical model that utilizes substantial preprocessing to improve performance, as well as using the COST-231 model for LOS links. In [234] the same authors propose heuristics to simplify the computational complexity of prediction by only calculating the most important ("dominant") paths. In [211], Sridhara *et al.* propose a ray-tracing approach, but only claim that its accuracy is sufficient for simulation (and not prediction). Finally, [105] provides a survey of various ray-tracing approaches. In addition to those papers published in the academic literature, there are also a number of commercial planning systems that provide similar prediction tools (e.g., [189, 239, 51]). The Remcom Wireless Insight software [189], for instance, packages a number of popular path loss prediction models discussed above with their own three-dimensional ray-tracing system.

The majority of recent work in this area is concerned with optimization and preprocessing to make feasible the intractable number of calculations required for this approach. Although in some ways these models are the most advanced of all the models on the table, they are not useful in practice for accuracy-sensitive coverage mapping because of their large computation and data requirements. Computing the many path loss estimates required to generate a coverage map for a large urban area in a reasonable amount of

time is simply outside the abilities of the current models. Those models that can compute results quickly do so by selecting a relatively small subset of rays to model, which may or may not be the most important. Precise two- and three-dimensional environmental vector data is seldom available, becomes stale quickly, and is often costly even when it is available. When this data is available, it is not clear which attributes are most important—in many scenarios, building materials (and their conductivity and permittivity properties) must also be known to make accurate predictions. In short, while these models offer a great deal of promise, there is still much work needed to understand their accuracy, and reduce the cost associated with their use (both in terms of time and data acquisition). In particular, developing an understanding of the relationship between the performance of these models to the fidelity of their input data is essential.

2.3 Modeling With Measurements

All of the preceding models discussed are *a priori*. They make predictions about a given network and a given environment either using analytical expectations about propagation or empirical models collected from a different (but hopefully similar) environment, or some combination thereof. The final category of models are those whose design is based on the assumption that there is no single set of *a priori* constants, functions, or data that allow for sufficient description of a new environment with sufficient accuracy. These models assume that the burden of making some number of measurements is unavoidable. In a sense, these are more than models—they define a method for collecting measurements (sampling strategy) and a means of predicting (interpolating) the values at locations that have not been measured.

The seminal work in this area is by W.C. Lee in [132]. In this work, Lee proposes a theoretically justified methodology for averaging signal strength. He suggests that a mobile receiver should make measurements in arcs at varying distances from the transmitter. He argues that measurements within 20 to 40 wavelengths of one another should be averaged to obtain a central tendency and that an appropriate sample size is at minimum, 36 measurements. For 2.4 GHz, this works out to between 0.625 and 1.25 m, which is in agreement with a study made by Shin, 25 years after Lee's original publication [208]. In this work, Shin does a measurement study of Institute of Electrical and Electronics Engineers (IEEE) 802.11b/g networks, attempting to model signal strength variation over small distances. He finds that the wideband modulation

schemes used in 802.11g result in some immunity to fast fading effects, and that small scale variations are "averaged out" within a radius of approximately 1 wavelength (3.1 mm for 2.4 GHz). He discovers that measurements have a strong spatial correlation within $\approx$ 1 m and become uncorrelated at larger distances. In [131] and [133], Lee expands his original measurement based work into a general purpose fitted model that is still commonly used in planning cellular networks.

In [73], Evans *et al.* utilize Lee's proposals to model the propagation of a transmitter at 1.9 GHz and find that they are able to achieve approximately 9 dB Root Mean Square Error (RMSE). A similar approach was also taken in [143], where Medeisis and Kajackas fit measurements to the Hata model and do some investigation of the number of measurements needed to sufficiently correct the model and appropriate measurement methods. They find that in their environment 15-20 measurements are needed to tune the model sufficiently, and that measurements are most useful when taken in clusters along a path. In [60] the authors explicitly fit measurements in their environment but fail to show significant improvement over *a priori* predictions (achieving, on average, 9 dB RMSE no matter the approach).

2.3.1 Explicit Mapping

Hills carried out some of the early high level work on formalizing wireless planning in his attempts to design a network for Carnegie Mellon University [95]. Based on his experiences, he would go on to develop a measurement apparatus for doing on-the-fly mapping of indoor propagation to aid in network planning [96]. The network engineer must place a temporary transmitter and roll the cart around collecting measurements. The cart counts wheel rotations to determine position and orientation within a building. The software on the cart plots signal strength measurements and can make suggestions about channel assignment to minimize interference with neighboring networks. In [76], Fretzagias and Papadopouli suggest a method for mapping indoor environments where the total area is divided into grid cells. A large number of nodes are used to sound the channel and make measurements. Then the measurements from each node are used in tournament/voting fashion to determine the average signal at each grid cell.

2.3.2 Partition Models

The next group of models worth describing are "partition based" models, where measurements are taken in an environment where the key obstructions are identified (i.e., walls, trees, buildings, etc.). In this approach, measurements are taken and static path loss values are fitted for each obstruction. Once the model is bootstrapped with these fits, it can be used (in theory) in other environments. An early example of this approach is in the very nice work by Durgin *et al.* in [64], where the authors study path loss in a suburban environment at 5.8 GHz. Naturally, this approach extends easily to indoor environments where there are a large number of explicit obstacles (walls). This approach has been investigated much more thoroughly by Rappaport and colleagues at various frequencies [205, 24]. In [249], Zvanovec *et al.* propose a similar model. However, due to the lack of substantive quantitative analysis in this paper it is difficult to draw strong conclusions from the results. In [243], Xiang *et al.* propose another partition-based model that also gives some attention to sampling. They propose a "lazy sampling" algorithm that greedily selects transmitter locations. A receiver is then used to make measurements on a regular grid and the measurements are used to train a partition model. The authors show that this approach can produce an interpolated coverage map with approximately 6 dB residual error.

2.3.3 Iterative Heuristic Refinement

The most recent active measurement model is that of Robinson *et al.* in [200]. In this work, the authors attempt to identify coverage holes in large wireless networks. They study the Technology For All (TFA) network operated by Rice and the Google WiFi network in Mountain View, California. Robinson's approach combines an *a priori* model with a fitted partition model and then uses a push-pull heuristic to make corrections from measurements. For a given Access Point (AP) node (n), and a given point (p), the SNR is predicted by:

$$P_{dB}(p,n) = P_0 - 10\alpha log\left(\frac{d(n,p)}{d_0}\right) + \beta(n,p) \tag{2.89}$$

where P_0 is the transmitter EIRP, $d(n,p)$ is distance from the point to the node, α is the path loss exponent,

d_0 is the reference distance, and $\beta(n, p)$ is a fitted offset function. Omitting the offset function, this equation is identical to equation 2.11 in section 2.2.1.2. The offset function makes use of a vector data terrain map that describes the types of buildings between an AP and each possible receiver site (pixel). A training phase determines the path loss per unit distance for each building type, which then informs the offset function:

$$\beta(n, p) = \sum_{f \in F} C_f \times w(n, p, f) \tag{2.90}$$

where $f \in F$ are the terrain "features" on the LOS path between the node n and point p, C_f is the fitted weight (i.e., path loss per unit distance) of the feature type f and $w(n, p, f)$ is the length of intersection between this feature and the line-of-sight path between n and p.

In Robinson's proposal, sufficient "pilot" measurements are made to determine the C_f values for all f and the environment wide α is determined. Then, this model is used to predict the signal strength of each AP to a large number of equally spaced points around the node. A coverage metric must be defined (e.g., $SNR > 20$), which says where a point is "covered" or not. By applying this metric to the predictions around the radius of a node the range of the node as a function of the azimuth angle is obtained. Robinson fits a step function to this curve and uses the number of segments in the fitted step function to create a "segmented" coverage prediction of each node with a relatively small number of segments. Figures 2.7 and 2.8 show an example of this sectorization and fitting.

The remainder of Robinson's method, involves iterative refinement. A measurement is made as close to each coverage boundary as possible and then the boundary is pushed or pulled by a constant amount. This process is repeated until the push/pull amount is less than some threshold (Robinson suggests 3 dB, which seems reasonable based on prior studies of expected repeated measures variance, e.g., [198]).

2.3.4 Active Learning and Geostatistics

As a generalization of the iterative refinement approach described above, the machine learning literature offers an approach called "active learning". In active learning systems, an algorithm is able to choose its training data, as opposed to passively trained systems that must learn a behavior from a set of "random" observations.

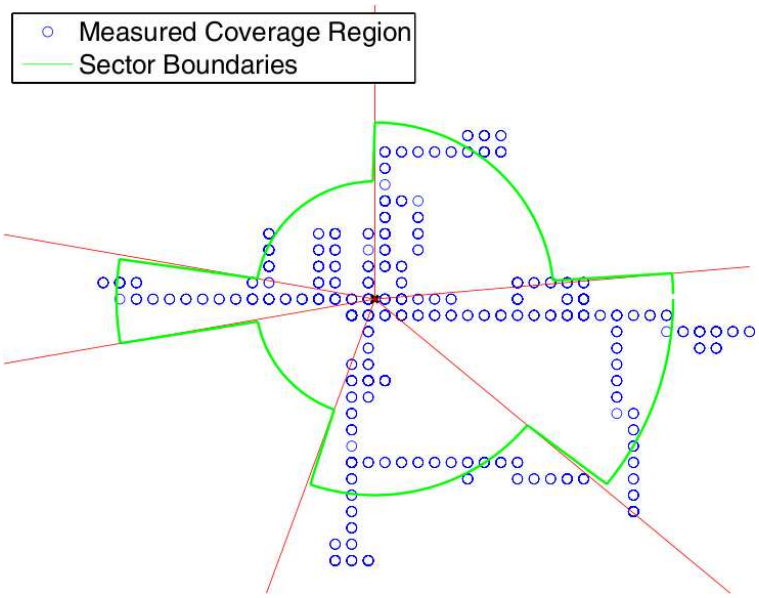


Figure 2.7: Example of sectorized propagation model for a single transmitter using the Robinson model. The measured (oracle) coverage is given as blue circles. The predicted/fitted coverage is given as sector boundaries that are adjusted (pushed and pulled) by additional measurements. Figure taken from [200].

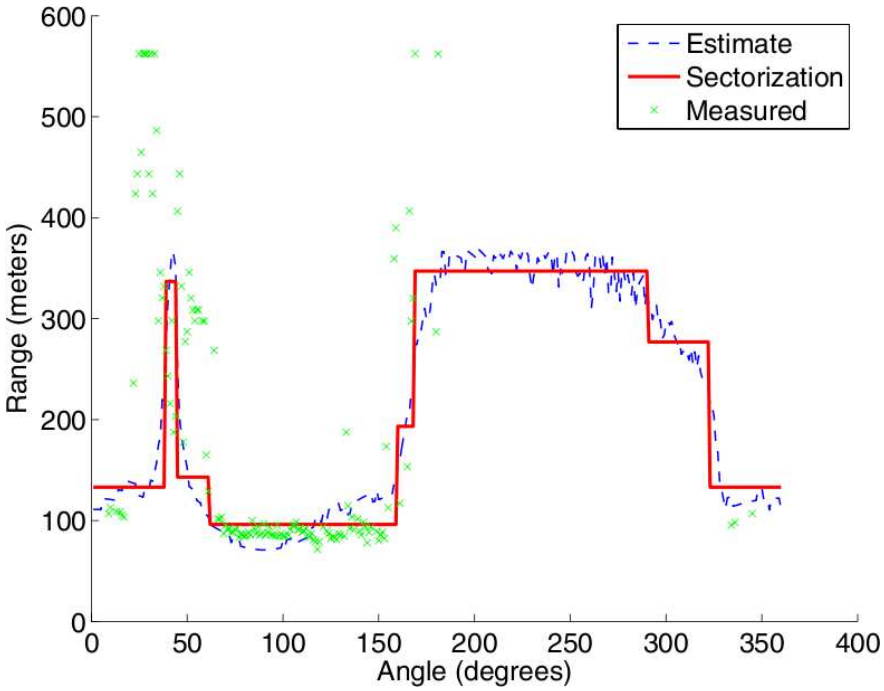


Figure 2.8: Example of fitted step function to measurements for the Robinson method. Figure taken from [200].

In [49], Cohn *et al.* provide a summary of this area, deriving active learning approaches to three kinds of learning systems: neural networks, mixed Gaussians, and locally weighted regression. Additional training data (samples) are chosen to minimize model variance. Cohn shows that active learning approaches far outperform randomly selected training data for training a model to solve the arm kinematics problem[2] .

Active learning has an analogous problem in the realm of geostatistics (and typically applied in ecological soil sampling) termed "optimized sampling" [230, 139]. In this version of the problem, additional data for a trained model is selected by minimizing some metric of variance (Kriging variance is generally used in geostatistical treatments). Regardless of the domain from which it is drawn, the task is fundamental: given some existing model, can we chose the next set of measurements that most improves the accuracy of the model itself?

2.4 Comparative Studies

The vast majority of existing work analyzing the efficacy of path loss models has been carried out by those authors who are proposing their own improved algorithm. In such cases, the authors often collect data in an environment of interest and then show that their model is better able to describe this data than one or two competing models. Unfortunately, this data is rarely published to the community, which makes comparative evaluations impossible. One noteworthy exception is the work of the COST-231 group in the early 1990's, which published a benchmark data set (900 MHz measurements taken in European cities) and produced a number of competing models that were well performing with respect to this reference [48]. This effort produced a number of well validated models that are tuned for 900 MHz transmitters in urban environments.

Similarly, there was substantial work done in the US, Japan, and several other countries in the 1960s and 1970s to come up with accurate models for predicting the propagation of analog TV signals (e.g., [57]). This flurry of work produced many of the models that are still used today in network simulators and wireless planning tools: the ITM [98], the Egli Model [69], and the Hata-Okumura model [157], to name a few. However, it is unclear what the implications are of using these models, which were created for use in a

[2] In the arm kinematics problem, a trained model attempts to predict the tip position of a robotic arm given a set of joint angles.

specific domain, to make predictions about another domain.

There are several works that compare a number of models with respect to some data. In [62], the authors compare five models with respect to data collected in rural and suburban environments with a mobile receiver at 910 MHz. They discuss the abilities of each model, but abstain from picking a winner. In [19], the authors compare three popular models to measurements collected at 3.5 GHz by comparing a least squares fit of measurements to model predictions. The authors highlight the best of the three, which turns out to be the ECC-33 model proposed in [72]. In [207], Sharma *et al.* do a very similar analysis, but instead focus on measurements made in India at 900 and 1800 MHz. In contrast to [19], they find that the Stanford University Interim model (SUI) and COST-231 models perform best.

2.5 Discussion

Making sense of the vast and varied landscape of path loss models can be a precarious task for the uninitiated researcher. In this chapter, a new taxonomy for reasoning about commonalities between these models was described. In terms of functionality and intent, the models can be further categorizd into classes based on those that are appropriate for (a) coverage and radio environment mapping, (b) rough planning, and (c) simulation. Applications that require accurate maps of the radio environment are probably best suited for an active measurement method that can resolve predictions with directed measurements. When it is not possible to make measurements of the environment directly, an experimenter must accept some (possibly substantial) error. Many-ray methods are promising, but their accuracy is intimately tied to the accuracy of data describing the environment and obstacles, which is seldom available at a useful resolution and can be very costly to collect and update. These models are also famously slow, requiring a substantial amount of computation for even a few predictions. Those looking to path loss models for rough planning are able to choose amongst dozens of seemingly similar proposals, accepting the caveat that it is impossible to verify accuracy. For this reason, the most-heavily used standard models are recommended for the sake of comparability (i.e., Okumura-Hata, Longley-Rice ITM, etc.). Simulations have similar needs to rough planning applications, except they also require the prediction of a distribution of reasonable values around the median for repeated-measured/Monte Carlo techniques. Hence, stochastic basic models (or deterministic

models with a stochastic fading parameter) are likely the most suitable, and there are several to chose from. Again, there is value in choosing amongst the most well-known, standard models (e.g., Hata with lognormal fading, or the recent Herring model).

Although there are many possible directions for future work in this area, measurement-based methods and rigorous (comparative) validation are most needed. Applications that make use of these models require an understanding of their real-world accuracy, and researchers need guidance in choosing amongst the many existing proposals. To this end, chapter 3 describes an important first step in this direction. Although this work will seek to provide a baseline performance for *a priori models*, more work is needed in general to resolve the imbalance between the quantity of models proposed and the extent to which they have been validated in practice.

Of all the models discussed so far, two extremes in terms of information requirements are apparant. On one end of the spectrum are basic models, like the Hata model, that require very little information about the environment—simply the link geometry and some notion of the general environmental category. At the other end are many-ray models which make use of vector data for obstacles to calculate specific interactions, requiring knowledge of the exact position and shape of all obstacles. In between these two extremes, there are very few models. Possible examples include the ITM and ITU-R 452 models, which make use of some additional information from public geographic datasets. A natural question then, is whether there is some other source of data available that could be used to inform better predictions, but is not as costly or difficult to obtain as detailed vector data. For instance: models that make use of high resolution satellite orthoimagery and machine vision techniques, a high resolution Digital Surface Model (DSM) (where surface clutter is not "smoothed away" as it is in digital elevation/terrain models, e.g., [103]), "crowd-sourced" building vector data *vis a vis* Google Sketchup [14], or topographic and zoning maps (e.g., [200]). So far, this data-mining approach to prediction, although promising, has seen little rigorous investigation.

There is simply no better way to generate truthful predictions than to start with ground-truth itself. For this reason, this thesis argues that the future of wireless path loss prediction methods will be active measurement designs that attempt to extract information from directed measurements. In particular, geostatistical approaches that favor robust sampling designs and explicitly model the spatial structure of measure-

ments are promising (e.g., [119, 236]). General machine learning approaches, and active learning strategies may also be fruitful, but applying those methods to the domain of path loss modeling and coverage mapping is currently unexplored. Future work in this area is likely to focus on refining sampling and learning strategies using measurement based methods, as well as extracting as much information as possible from existing sources using data mining. Methods for parallelizing computation and preprocessing datasets are also needed to make predictions quickly (this is especially true when these models are used in real time applications). And, once predictions are made, efficient storage and querying of these spatial databases is an opportune area for further work.

As the the prevalence and importance of wireless networks continues to grow, so too will the need for better methods of modeling and measuring wireless signal propagation. This chapter has given a broad overview of approaches to solving this problem proposed in the last 60 years. Most of this work has been dominated by models that extend on the basic electromagnetic principles of attenuation with theoretical and empirical corrections. More recently, work has focused on developing complex theoretical deterministic models. It is likely that the next generation of models will be data-centric, deriving insight from directed measurements and possibly using hybridized prediction techniques, such as the geostatistical approach described in this thesis. Regardless of the approach that is taken, there is substantial possibility for future work in this area, with the promise of great impact in many crucial applications.

Chapter 3

Bounding the Error of Path Loss Prediction

Despite the large quantity of work done on modeling path loss, there is an important shortcoming that this chapter begins to address: there have been relatively few comparative evaluations of path loss prediction models using a sufficiently representative data set as a basis for evaluation. Those studies that do exist only make comparisons between a small number of similar models. And, where there has been substantial work of serious rigor done, for instance in the VHF bands where solid work in the 1960's produced well validated results for analog television (TV) propagation, it is not clear how well these models work for predicting propagation in different types of systems operating at different frequencies. The result is that wireless researchers are left without proper guidance in picking among dozens of propagation models. Further, among the available models it is not clear which is best or what the penalty is of using a model outside of its intended coverage. In [44], for instance, Camp *et al.* show that a wireless mesh network planned with a given path loss model can be massively under- or overprovisioned as a result of small changes to model parameters. For the purpose of this thesis, it is crucial to put practical bounds on the performance of existing methods in order to define a clear benchmark of success.

This chapter analyzes 30 propagation models spanning 65 years of publications using five novel metrics to gauge performance. Although many of these models are quite different from one another, they all make use of the same basic variables on which to base their predictions: position (including height and orientation) of the transmitter and receiver, carrier frequency, and digital elevation model and land cover classification along the main line-of-sight (LOS) transmit path. These models utilize a mix of approaches:

[0] Work in this chapter has appeared in [173, 174, 167, 172]. Data collected for the experiments in this chapter has been made publicly available at [166, 170].

empirical, (purely) analytical, stochastic or some combination thereof. They are tested in this analysis without starting bias as to which should peform best. Active-measurement models (e.g., [200] and the geostatistical approach advocated by this thesis), which make use of directed *in situ* measurements to correct their predictions are not considered here, as they are the focus of later chapters in the thesis.

The focus in this chapter is the efficacy of the models studied at the task of *predicting* median path loss values in environments with representative terrain and a large range of equipment and link lengths. Many authors have considered the problem of predicting outdoor path loss in uncluttered environments to be solved. We will see this is far from true— making accurate *a priori* predictions about path loss, without *in situ* measurements, with the models available, is a very difficult task even in "simple" environments.

In the end, the results show that no single model is able to predict path loss consistently well. Even for the seemingly simple case of long links between well-positioned antennas in a rural environment, the available models are unable to predict path loss at an accuracy that is usable for any more than crude estimates. Indeed, no model is able to achieve a RMSE of less than 14 dB in rural environments and 8-9 dB in urban environments—a performance that is only achieved after substantial fine tuning. Explicit data-fitting approaches do not perform better, producing 8-9 dB RMSE as well. This conclusion motivates the work on more rigorous measurement based approaches that forms the remainder of this thesis.

3.1 Measurement

This section describes data sets collected to address the goals of this chapter. These measurements were collected over the course of several years in multiple environments and with differing (but consistent) hardware. They range from "clean" measurements taken in rural New Zealand, to "noisy" measurements collected in the urban center of a large US city along with some special measurements to investigate points of particular interest, such as measurements with phased array and directional antennas, and some in suburban environments. Overall, these data sets combine to paint a unique picture of the real-world wireless radio environment at varying levels of complexity. Table 3.1 provides a summary of these data sets.

Campaign	Name	Environment	Type	Frequency	Method	Transmitters	Measurements
A	wart	Campus	Point-to-Point	2.4 GHz	Packet	7	33,881
A	wart/snow	Campus	Point-to-Point	2.4 GHz	Packet	7	24,867
B	pdx	Urban	Urban Mesh/Infrastructure	2.4 GHz	Packet	250	$\approx$ 117
B	pdx/stumble	Urban	Urban Mesh/Infrastructure	2.4 GHz	Packet	59,131	200,694
C	boulder/ptg	Campus	Infrastructure/Downstream	2.4 GHz	Packet	1,693	1,693
C	boulder/gtp	Campus	Infrastructure/Upstream	2.4 GHz	Packet	329	329
D	cost231	Urban	Infrastructure/Downstream	900 MHz	Continuous Wave (CW)	2,336	2,336
E	wmp/a	Rural	Point-to-Point/Infrastructure	5.8 GHz	Packet	368	2,090,943
E	wmp/g	Rural	Point-to-Point/Infrastructure	2.4 GHz	Packet	368	20,314,594
F	tfa	Suburban	Mesh/Infrastructure	2.4 GHz	Packet	22	389,401
G	google	Urban/Suburban	Mesh/Infrastructure	2.4 GHz	Packet	168	75,101

Table 3.1: Summary of data sets

3.1.1 Packet-Based Measurements

With the exception of the COST-231 data, discussed in section 3.1.3.3 below, all data sets used in this thesis were collected using commodity hardware and packet-based measurements were used to determine received signal strength. This approach differs from some prior work on path loss modeling that uses continuous wave (CW) measurements [91, 48]. When using packet-based methods to collect information about received signal strengh and path loss, a transmitter is configured to transmit "beacon" frames periodically. A (often mobile) receiver records these beacon frames. Using an open source driver, such as MadWifi [12], and a compatable chipset, frames can be recorded in their entirety to the harddisk in real-time using any number of userspace software tools (e.g., tcpdump). If these frames are recorded with the optional Radiotap header [8] (or equivalently, the more archaic Prism II header) then the record will include information about the physical layer, such as the received signal strength of the frame, any Frame Check Sequence (FCS) errors, and a noise floor measurement. Using this approach, inexpensive commodity hardware can be used to make extensive passive measurements of a wireless network.

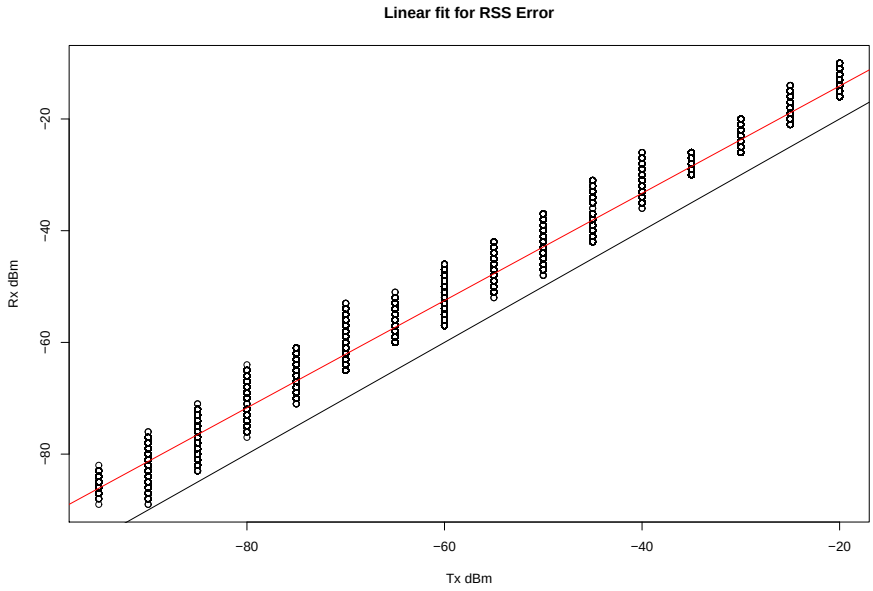


Figure 3.1: Linear fit to RSS error observed from commodity cards during calibration.

To get an idea of how accurate commodity radios are in measuring Received Signal Strength (RSS),

some calibration experiments were performed in a conductive setting. Each of four radio cards was directly connected to an Agilent E4438C Vector Signal Generator (VSG). The cards were all Atheros-based Lenovo-rebranded Mini-PCI Express, of the same family (brand and model line) chipset to those used for all of our packet-based measurements. The VSG was configured to generate 802.11 frames and the laptop to receive them. For each of the four cards many samples were collected while varying the transmit power of the VSG between -20 dBm and -95 dBm (lower than the receive sensitivity threshold of just about any commodity 802.11 radio) on 5 dB increments. Finally, a linear least squares fit was performed, finding a slope of 0.9602 and adjusted R-squared value of 0.9894 (indicating a strong fit to the data). Figure 3.1 shows this data and the fit line. The commodity radios perform remarkably well in terms of RSS measurement. To correct for the minor error they do exhibit, the slope of this fit can be used to adjust our measurements, dividing each measurement by the slope value.

However, there is a drawback to this approach. Packet-based methods necessarily "drop" measurements for packets that cannot be demodulated. All receivers have fundamental limits in their receive sensitivity that are a function of their design. However, because packet-based measurement techniques rely on demodulation of packets to determine the received signal strength, they have a necessarily lower sensitivity than receivers that calculate received power from pure signal (i.e., continuous wave measurements). Additionally, without driver modification, commodity receivers generally update noise floor measurements infrequently. For the purpose of analyzing accuracy of median path loss prediction (as is done in chapter 3), these limitations are not problematic. In one sense, commodity hardware "loses" only the least interesting measurements—if we are unable to decode the signal at a given point, we are at least aware that the signal is *below* the minimum detectable signal for basic modulation schemes, and is as a result, unlikely to be usable for many applications.

It should be noted that packet-based measurement methods are not appropriate for all modeling tasks—the tradeoff between convenience and affordability of commodity hardware versus the completeness of the measurements must be considered. For instance, if the goal of a measurement campaign is to sense signals or interference near the noise floor in order to predict capacity for next generation protocols, or if the goal is to model delay spread or Doppler shift, then packet-based measurements will not be suffi-

cient. However, the work in this chapter has less demanding data requirements than these applications. For the purpose of measuring median SNR at a given point in space from the perspective of a typical receiver, packet-based measurements made with commodity hardware are both sufficiently accurate and convincingly representative.

3.1.2 Rural Measurements

In cooperation with the Waikato Applied Network Dynamics (WAND) research group at the University of Waikato [13] and the RuralLink wireless internet service provider (WISP) [9], a large set of measurements was acquired from a commercial network in rural New Zealand. These measurements were collected for the Wireless Measurement Project (WMP) [185]. Rural environments are simpler both in the sense that there are fewer obstacles to cause fading, and those obstacles that do exist are typically large and constant (e.g., mountains and terrain features) which produce only large scale shadowing and minimal small scale (fast) fading. Moreover, the isolated nature of rural networks result in less interference from neighboring competing networks, which can create random fades that are difficult to predict and model. Hence, the measurements here are intended to form a comparative baseline for the measurements in more complex environments.

The network used in this study is a large commercial network that provides Internet access to rural segments of the Waikato region in New Zealand (as well as some in other regions). The overall approach to measurement involves periodically broadcasting measurement frames from all nodes and meanwhile recording any overheard measurement frames. Every two minutes, each device on the network transmits a measurement frame at each supported bitrate. Meanwhile, each device uses a monitor mode interface to log packets. Because this is a production network, privacy concerns are of clear importance, which is why all measurements are made with injected packets and a Nondisclosure Agreement (NDA) was required for use of parts of the data that contained sensitive information (principally client locations).

The network is arranged in the typical hub and spoke topology as can be seen in figure 3.2. The backhaul network is composed of long distance 802.11a links operating at 5.8 GHz. Atypically liberal power regulations in New Zealand and Australia around 5.8 GHz allow for much longer links than can be seen in

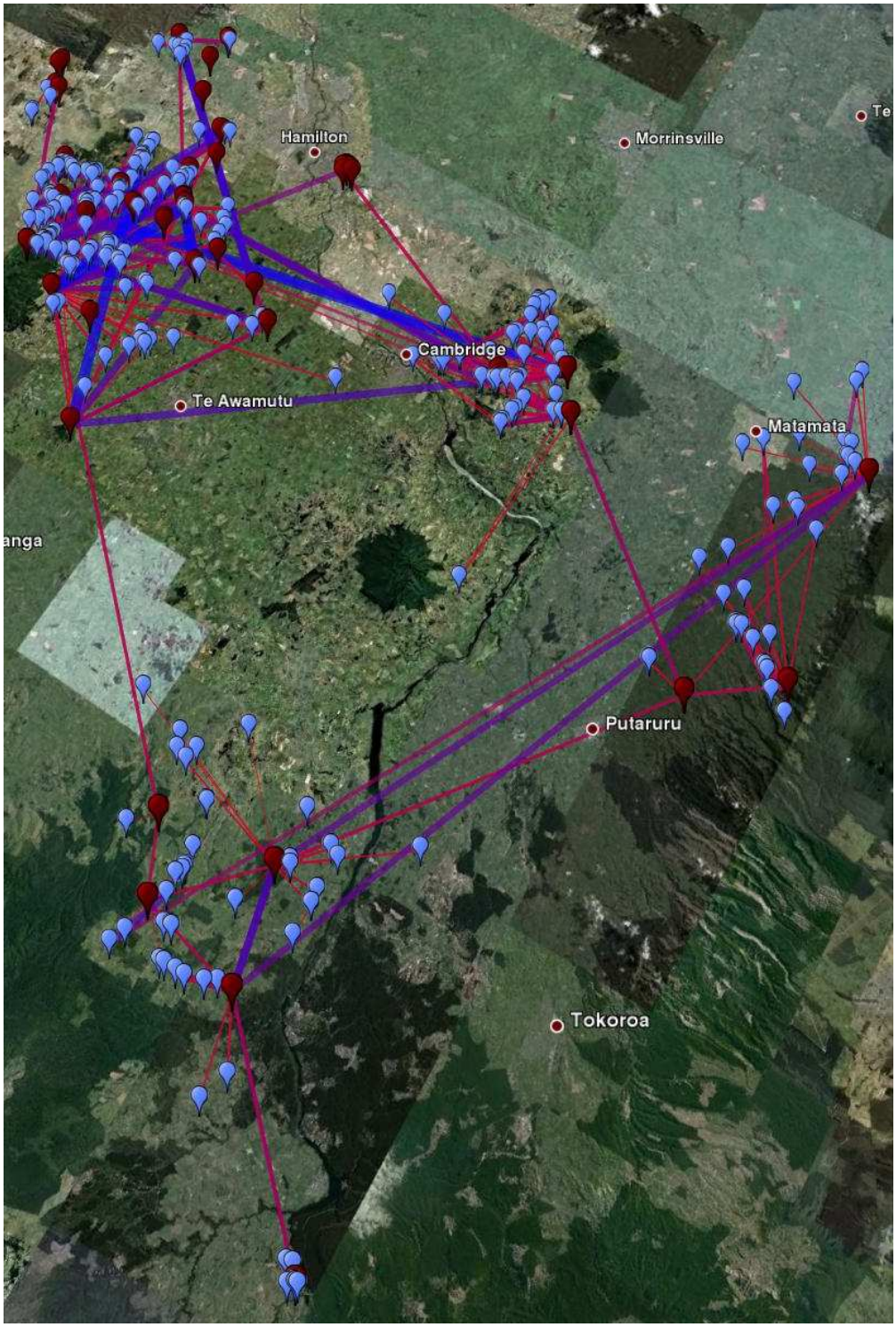


Figure 3.2: The largest of three disconnected sections of the network (80x100km). Link color indicates strength: blue implies strong, red implies weak. Backhaul nodes (mainly 5.8 GHz) are red and CPEs are light blue.

most other places in the world—40 km is a typical link length in this network[1]. These are commonly point-to-point links that use highly directional antennas that are carefully steered. The local access network is composed of predominantly 802.11b/g links that provide connectivity to Client Premises Equipment (CPE). Often, an 802.11g AP with an omnidirectional or sector antenna will provide access to a dozen or more CPE devices that have directional (patch panel) antennas pointing back to the AP. With few exceptions, each node in the network is an embedded computer running the Linux operating system that allows us to use standard open source tools to perform measurement and monitoring. All nodes under measurement use an Atheros-brand radio and the MadWifi driver [12] is used to collect frames in monitor mode and record received signal strengths using the radiotap extension to libpcap [8].

After collection, the data requires scrubbing to discard frames that have arrived with errors. Because there is substantial redundancy in measurements (many measurements are made between every pair of participating nodes), discarding some small fraction of (presumably randomly) damaged frames is unlikely to harm the integrity of the data overall. As a rule, any frame that arrives with its checksum in error or those from a source that produces less than 100 packets is discarded. For the work in this thesis, one representative week of data collected between July 25th, 2010 and August 2nd, 2010 is used. Because detailed documentation about each node simply did not exist, some assumptions were made for analysis. The locations of nodes for which there is no specific GPS reading are either hand coded, or in the case of some client devices, geocoded using an address. Antenna orientations for directional antennas are assumed to be ideal—pointing in the exact bearing of their mate. All nodes are assumed to be positioned 3 m off the ground, which is correct for the vast majority of nodes. While these assumptions are not perfect, and are clearly a source of error, they are reasonably accurate for a network of this size and complexity. Certainly, any errors in antenna heights, locations, or orientations are on the same scale as those errors would be for anyone using one of the propagation models analyzed to make predictions about their own network of interest.

In the end, the scrubbed data for a single week constitutes 19,235,611 measurements taken on 1,328 links (1,262 802.11b/g links at 2.4 GHz and 464 802.11a links at 5.8 GHz) from 368 participating nodes.

[1] Fixed radio links (Unlicensed National Information Infrastructure (U-NII) devices) operating between 5.725 and 5.825 GHz that use wide band digital modulation are allowed an EIRP of 200 W [1].

Of these nodes, the vast majority are clients and hence many of the antennas are of the patch panel variety (70%). Of the remaining 30%, 21% are highly directional point-to-point parabolic dishes, and 4.5% each of omnidirectional and sector antennas.

3.1.3 Urban Measurements

In addition to the "baseline" measurements in a rural setting, measurements were collected in three additional environments to complete the picture of the urban/suburban wireless propagation environment. Figure 3.3 provides a schematic of the three urban data sets and table 3.1 provides further details. The three campaigns cover the three transceiver configurations that are most important in the urban wireless environment. The first, **A**, concerns well-positioned (i.e., tower or rooftop) fixed wireless transceivers. This sort of link is typically used for backhaul or long distance connections (e.g., [20]). The second, **B**, concerns propagation between a single fixed ground-level node (i.e., on a utility pole) and mobile ground-level client devices. Finally, **C**, concerns infrastructure network configurations where one fixed well-positioned transmitter (AP) is responsible for serving multiple ground-level mobile nodes.

3.1.3.1 Backhaul

The first data set, **A**, was collected using the University of Colorado at Boulder (CU) Wide Area Radio Testbed (WART), which is composed of six 8-element uniform circular phased array antennas [29]. Figure 3.4 shows the layout of this testbed. The devices are mounted on rooftops on the CU campus and in the surrounding city of Boulder, Colorado. These devices can electronically change their antenna pattern, which allows for them to operate as a directional wireless network with a main lobe pointed in one of 16 directions or as an omnidirectional antenna whose gain is (approximately) uniform in the azimuth plane. To collect this data, an "NxN scan" is done of the sort proposed in [41], which results in RSS measurements for every combination of transmitter, receiver, and antenna pattern. In short, this works by having each AP take a turn transmitting in each state while all other nodes listen and log packets. Identical measurements were collected during the winter (no leaves), during a snowstorm, and during the summer of 2010. These network measurements are applicable to rooftop-to-rooftop communication systems, including cell networks, and

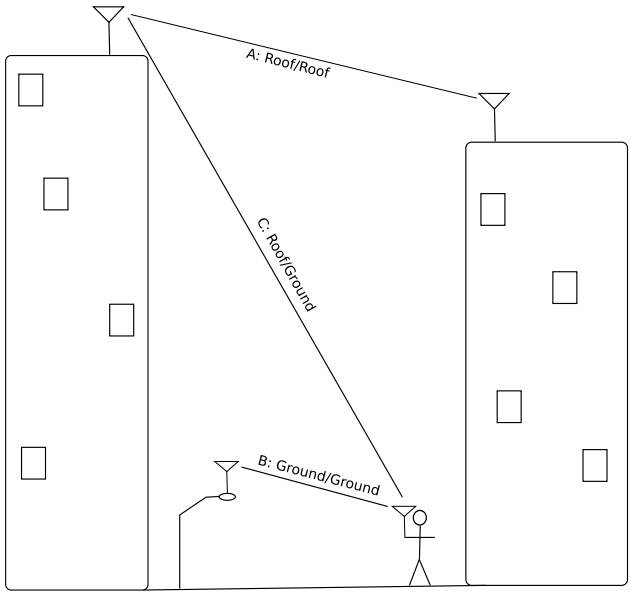


Figure 3.3: Visual schematic of three urban data sets. A: roof to roof measurements from CU WART (Wide Area Radio Testbed), B: ground (utility poles) to ground (mobile node) measurements in Portland, Oregon, C: roof to ground and ground to roof measurements from CU WART.

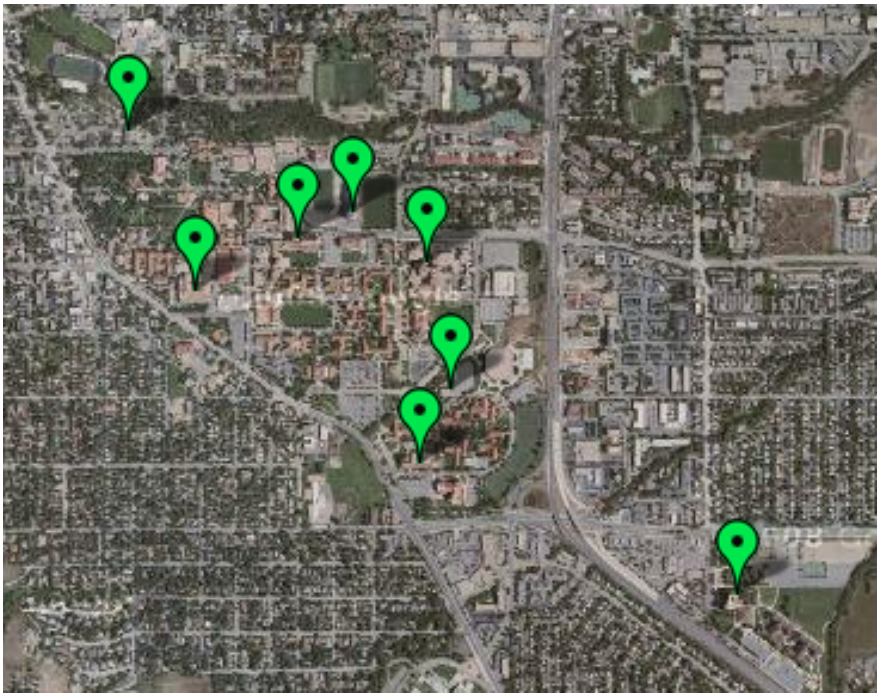


Figure 3.4: University of Colorado Wide Area Radio Testbed (CU-WART)

point-to-point or point-to-multipoint wireless backhaul networks both with directional antennas and with omnidirectional antennas. Although this is a reasonably small network, the representativeness of the environment (a typical urban/suburban campus) and the large number of effective antenna patterns (17^6 unique combinations) that can be tested provide a strong argument for the generalizability of this data.

3.1.3.2 Street Level Infrastructure

The second set of measurements, **B**, involves three data sets from three urban municipal wireless networks: a (now defunct) municipal wireless mesh network in Portland, Oregon, the Google WiFi network in Mountain View, California, and the TFA network in Houston, Texas. All three data sets involve data collected with a mobile client. As a standard practice, the precision of the GPS coordinates is truncated to five significant digits, which has the effect of averaging measurements within a 0.74m (≈ 6 wavelength) circle (a conservative averaging by the standard of [132]).

Portland, Oregon

In this network, 70 APs are deployed on utility poles in a 2 km by 2 km square region. Each AP has a 7.4 dBi omnidirectional antenna that provides local coverage in infrastructure mode. These measurements were collected during the summer of 2007. This data set, which consists of both laborious point testing and extensive war-driving data is most representative of ground-to-ground links in urban environments. The data collection method for this data set is outlined in section 4.2. In short, collection involved a two-stage process. First, a mobile receiver was driven on all publicly accessible streets in the 2 km by 2 km region. The receiver was a Netgear WGT-634u wireless router running OpenWRT linux [17] and the open-source sniffing tool Kismet [15]. The Kismet tool performs channel-hopping to record measurements on all 11 802.11b/g channels which imposes a uniform random sampling (in time) on the observed measurements. The receiver's radio is a Atheros-brand chipset, with an external 5 dBi magnetic roof-mount "rubber duck" antenna and a Universal Serial Bus (USB) GPS receiver. Passive measurements of management frames (beacons) were recorded to a USB compact flash dongle. This results in a large set of measurements that is referred to as "pdx/stumble" here. After this initial stage, 250 additional locations were selected at random from within the region and tested more rigorously with a state-based point tester. At each of these points

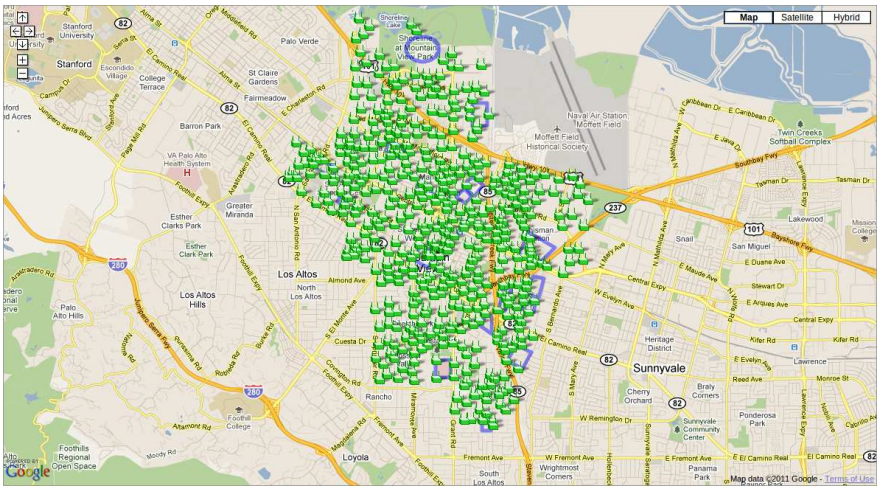


Figure 3.5: Google WiFi Network in Mountain View, California

physical layer information was recorded (i.e., SNR) along with results from higher layer tests. This smaller data set is called “pdx” in the remainder of the thesis and the data collection procedure is described in more detail in section 4.2.1.

Mountain View, California

The Google WiFi network [83], deployed in Mountain View, California covers much of the city (31 km^2) with 447 Tropos-brand [226] 2.4 GHz 802.11 mesh routers. Figure 3.5 provides a basic layout of the network and gives an idea of the extent and density of the deployment. The measurements used here were collected by Robinson *et al.* between October 3rd and 10th in 2007 for their work in [200]. These measurements were made publicly available at [191] and involve passive measurements over a subset of the coverage area (12 km^2) encompassing 168 mesh nodes. These nodes are mounted on light poles as in the Portland measurements and have a 7.4 dBi omnidirectional antemnna for local coverage in addition to the backhaul network. The measurements were made with an IBM T42 laptop with a 3 dBi antenna and GPS receiver running the NetStumbler sniffing software [16]. As with the Portland measurements, these are all passive measurements of management frames (beacons) and the sniffer employs channel-hopping to make a uniform random sample (in time) of all 11 channels. The Received Signal Strength Indicator (RSSI) and noise values are recorded for each packet overheard along with a time-stamp and GPS location. Some

minor anonymization of the data has been done to remove unique identifiers (Basic Service Set Identifier (BSSID)s). RSSI is converted to RSS by subtracting 149 from each value [199]. Precise height and transmit power control information was not recorded for this data, so in our application we use the reasonable constant values of 20 dB (100 mW) transmit power (as extracted from Tropos product white-paper specificiations) and 12m for the utility pole height.

Houston, Texas

The final set of street level infrastructure measurements comes from the community wireless mesh network constructed by Rice University and the TFA non-profit organization in Houston, Texas [222]. Figure 3.6 shows a heatmap of the measurements. These measurements were collected by Robinson, Camp *et al.* for their work in [44] and [200]. The measurements have been made publicly available at [192]. This network involves 18 wireless nodes in a residential area in Southeast Houston, providing coverage to approximately 3 km^2 and more than 4,000 users. In the data collection, the NetStumbler software was used on a laptop with an a GPS device and Orinoco Gold 802.11b wireless interface (Atheros chipset) connected to a car-roof mounted 7 dBi omnidirectional antenna. As with the other measurements, all data collection is passive and the software channel-hops to record a random sample of overheard management frames (beacons) on each of the 11 channels. The drive-test covers all city streets in the region and was carried out 15 times between the hours of 10am and 6pm between December 15th 2006 and February 15th, 2007. Although this is a winter data collection, Houston has a tropical climate, so it is presumed that the fading due to foilage is constant throughout the year. The measurements contain signal strength, noise, and location values as well as the vehicle's average velocity at the point of measurement.

3.1.3.3 Wide Area Infrastructure

The final data set, **C**, involves two sets of measurements: one carried out at CU of the WART and one set of published measurements from a well-placed transmitter in Munich, Germany.

Boulder, Colorado

The first data set was collected using a mobile node (a Samsung brand "netbook") with a pair of diversity antennas. In this experiment, the six rooftop CU WART nodes were configured to transmit 80

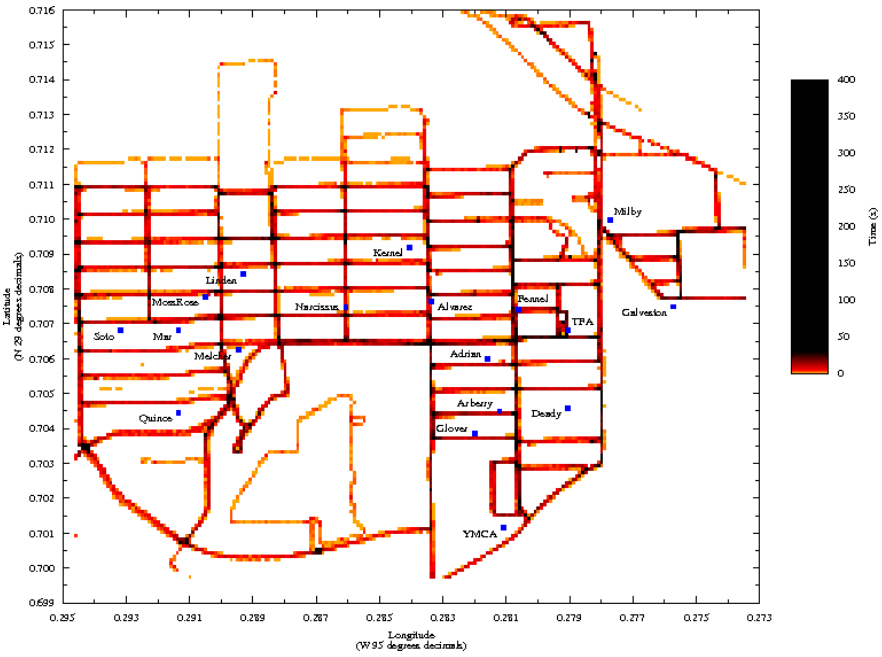


Figure 3.6: TFA-Wireless Network measurements in Houston, Texas

byte “beacon” packets every $0.5 + U(0.0, 0.5)$ seconds, where $U(X, Y)$ is a uniformly distributed random number between X and Y. Beacons are configured to transmit at 1 Mbps, so that possible effects of Doppler spread on higher datarate waveforms are avoided. Similarly, the mobile device was configured to transmit beacons at the same rate. Meanwhile, each rooftop testbed node was configured to its 9 dBi omnidirectional antenna pattern.

All nodes, including the mobile node, were configured to log packets using a second monitor mode (promiscuous) wireless interface. The mobile node was additionally instrumented with a USB GPS receiver that was used both to keep a log of position and to synchronize the system clock so that the wireless trace was in sync with the GPS position log. These measurements were collected during the summer of 2010. During the experiment, the mobile node was attached to an elevated (nonconducting) platform on the front of a bicycle. The bicycle was pedaled around the CU campus on pedestrian paths, streets, and in parking lots. This data set is most representative of an infrastructure wireless networks where a well-positioned static transmitter must serve mobile clients on the ground. This data set is subdivided into the upstream part (“boulder/gtp”) and the downstream part (“boulder/ptg”).

Munich, Germany

The second group of measurements is from a reference data set collected by the COST-231 group at 900 MHz [48] in Munich in 1996. This data set, which provides path loss measurements collected by a mobile receiver from three well-placed (rooftop) transmitters is closest in intent to our data set **C**, but does not include upstream measurements.

3.2 Implementation Details

Table 2.1 in section 2.2 provides details of the models evaluated in this study. Each of the 30 models is implemented from their respective publications in the ruby programming language. Section C.2 in appendix C provides the source code for the implementations. Only one of the models, the ITM [98], has a reference implementation. Hence, there are fundamental concerns about correctness. To address this basic issue, sanity checking of model output is performed. However, without access to the data sets on which the models were derived, or their reference implementations, it is impossible to make a more rigorous verification than

this.

3.2.1 Terrain Databases

Terrain Models require access to a DEM, and in the case of ITU-R 452, a Landcover Classification Database (LCDB) as well. The DEM used for the networks in the United States is a publicly available raster data set from the United States Geological Survey (USGS) Seamless Map Server, providing 1/3 arcsecond spatial resolution. The US LCDB is also provided by the USGS as a raster data set, which is generated by the USGS using a trained decision tree algorithm. For the New Zealand data sets, DEM and LCDB data are provided by the Environment Waikato organization. The DEM has a vertical precision of 1 m and an estimated accuracy of 5-6 m RMSE. The GDAL library [75] is used to perform coordinate conversions and data extraction to generate path profiles for the terrain algorithms.

3.2.2 Corrections for Hata-Okumura

In the implmentation of Hata-Okumura used in this analysis, and its derivative models, a few crude corrections are made to antenna heights in the event that they fall outside of the models' coverage (and would therefore produce anomalous results). First, the minimum of the two heights is subtracted from both so that they are relative. For instance, antenna heights of 30 and 40 m become 0 and 10. Then, heights are swapped if necessary so that the transmitter height is always higher than the receiver height (at this point the receiver height will be zero). Next, one meter is added to the receiver height and subtracted from the transmitter height, keeping the relative difference but setting the receiver height to 1 m. For instance 0 and 10 m would become 1 and 11 m. Finally, the transmitter height is decreased or increased as necessary so that it is above the minimum (30 m) and below the maximum (200 m) permissible values for the Hata-Okumura model.

These corrections are necessary to use the Hata-Okumura model with transmitter or receiver heights that would otherwise produce meaningless (infinite) results. It is not certain what the impact is on the model performance by making these corrections. However, it stands to reason that even if the performance is negatively impacted, an inaccurate prediction will still be closer to the true answer than an anomalous

(infinite) prediction.

3.3 Method

The approach taken is to ask each model to offer a prediction of median path loss for each link in the data. The model is fed whatever information it requires, including DEM and LCDB information. The model produces an estimate of the loss $\hat{L}$ that is combined with known values to calculate the predicted received signal strength P_r:

$$P_r = P_t + G_t(\theta) + G_r(\phi) - \hat{L} \tag{3.1}$$

Where G_t is the antenna gain of the transmitter in the azimuthal direction (θ) of the receiver and G_r is the antenna gain of the receiver the azimuthal direction (ϕ) of the transmitter. These gains are drawn from measured antenna patterns. The antenna patterns were derived for each antenna empirically, using the procedure described in appendix A. The transmit power (P_t) is set to 18 dBm for all nodes, which is the maximum transmit power of the Atheros radios that all measurement nodes use. For a given link, the median received signal strength value is calculated across all measurements ($\bar{P_r}$). Then, the prediction error, ϵ, is the difference between this prediction and the median measured value:

$$\epsilon = \bar{P_r} - P_r \tag{3.2}$$

Some models come with tunable parameters of varying esotericism. For these models, a range of reasonable parameter values are tried without bias towards those expected to perform best.

This entire process requires a substantial amount of computation, but is trivially parallelizable. To make the computation of results tractable, the task of prediction is subdivided into a large number of simultaneously executing threads and the results are merged upon completion. Figure 3.7 shows a schematic of the process. Parallel computation must occur in two sequential stages. During the first stage, path profile information is extracted and prepared for each link in parallel, and during the second stage this information is fed to each algorithm for each link, which can also be done in parallel. With the merged data in hand, each

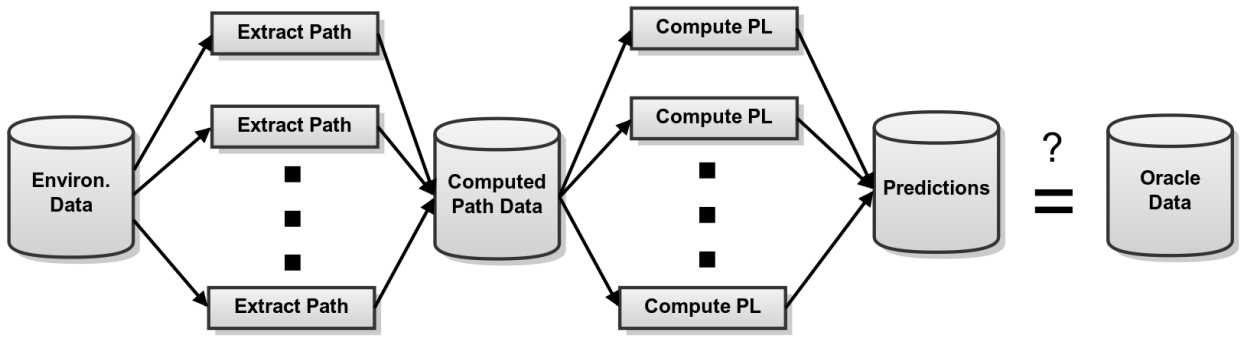


Figure 3.7: Schematic showing the process of parallel computation of path loss predictions using many models and many links.

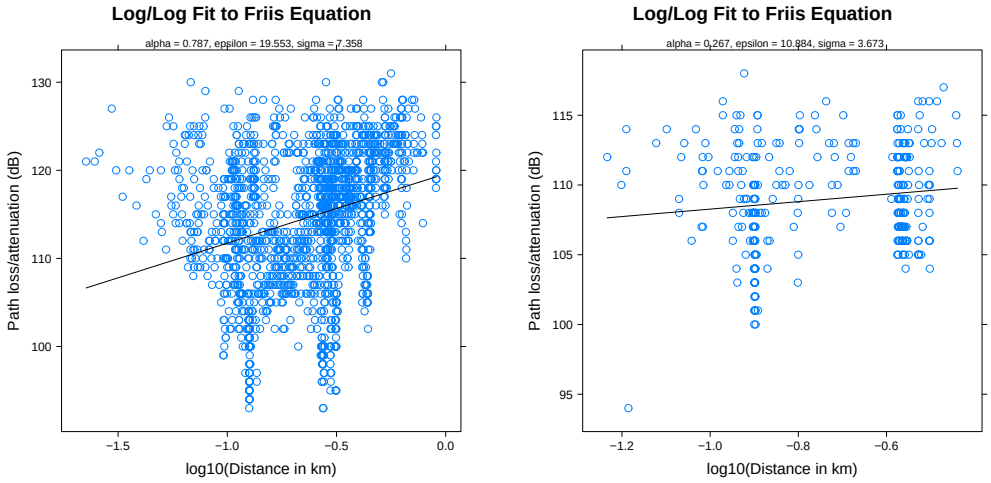
prediction is compared with an oracle value for the link. This oracle value is computed from the measured received signal strength for the link as well as known values for the transmitter power and antenna gain.

It is worth noting that among the models studied, only very few were designed with the exact sort of networks studied here in mind. Indeed, some are very specific about the type of environment in which they are to be used. In this study both appropriate and "inappropriate" models are given an equal chance at making predictions for our network—there is no starting bias about which should perform best.

The next section describes the process of explicitly fitting the data to a theoretical model and looking at the number of measurements required for a fit. This gives an initial estimate of expected error for direct (naïve) fits to the collected data. Then, to analyze the performance of the algorithms, five domain-appropriate metrics of decreasing stringency are proposed. The performance of each model with respect to these metrics, as well as general trends and possible sources of systematic error, are described in section 3.6. Finally, in an attempt to put a lower bound on model error, explicit parameter fitting of the best models is performed and this best-case performance is compared to the naïve approach of straight line fitting.

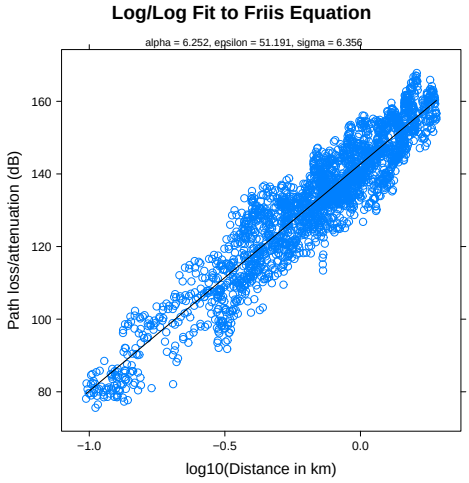
3.4 Simple Log-Domain Data Fitting

Consider equation 2.8 in section 2, which describes the fundamental power law relationship between path loss and distance. It is common in the literature to show this relationship as a straight line on a log/log plot. When this equation is modified to have a flexible exponent and error term, it is possible to do a linear

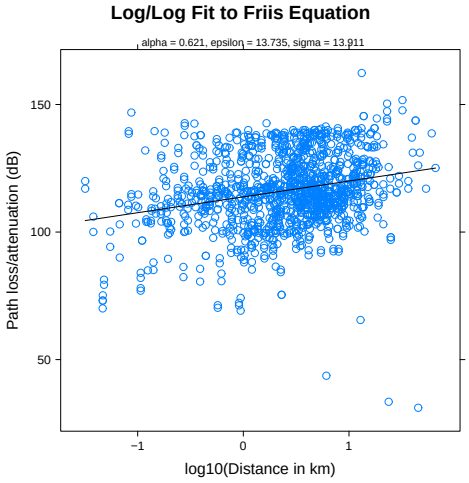


(a) Boulder Downstream

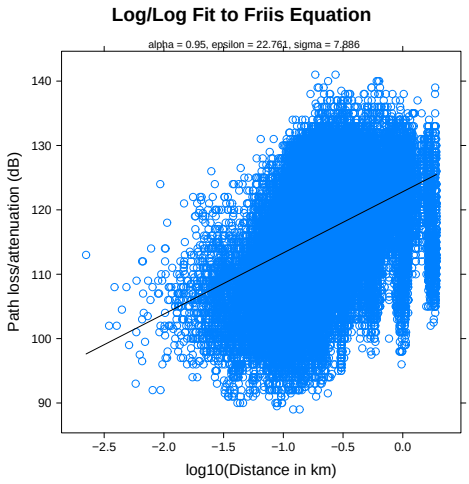
(b) Boulder Upsteam



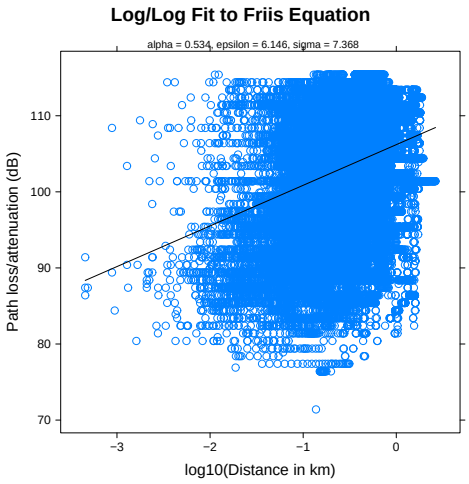
(c) COST-231



(d) WMP

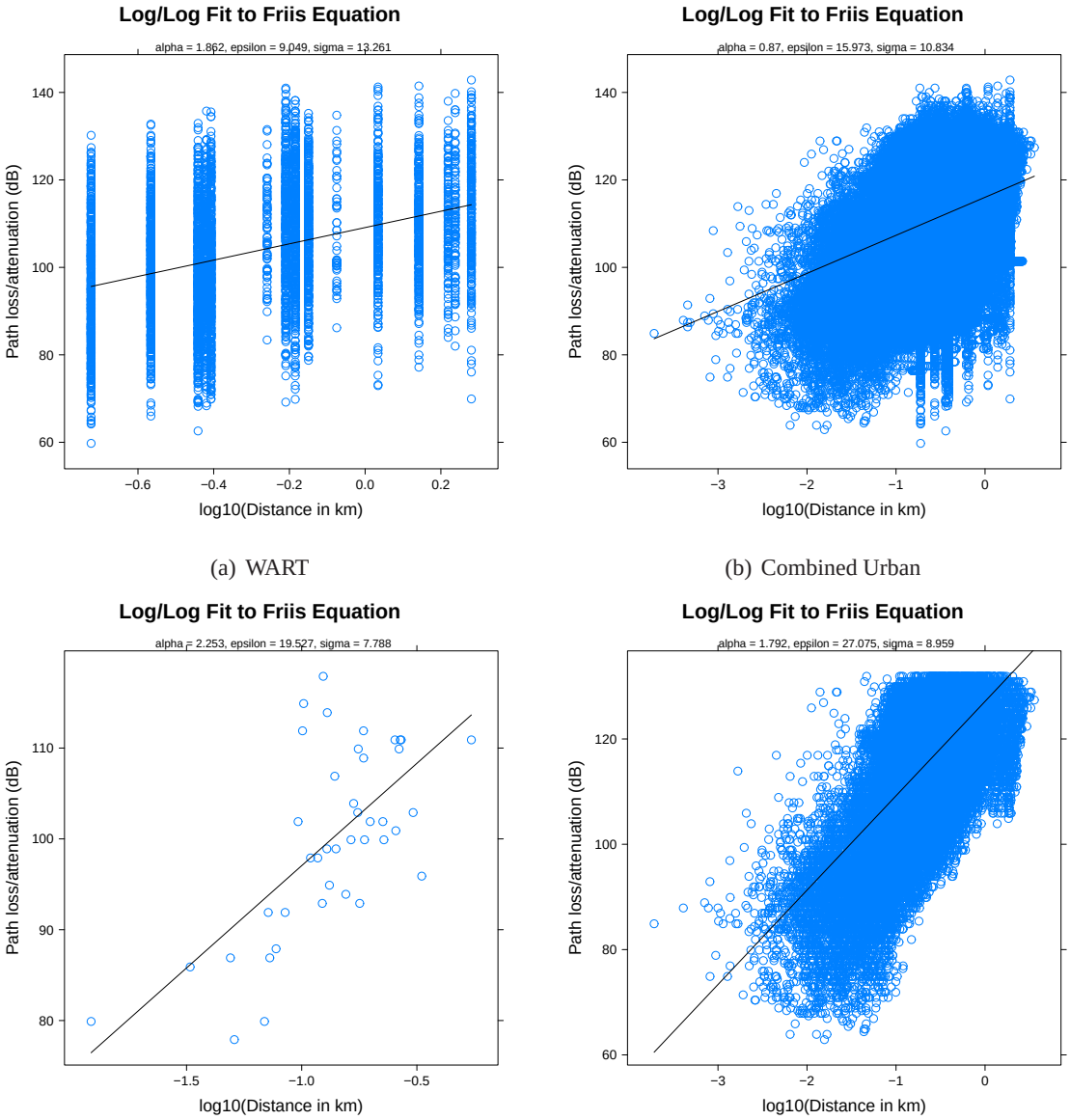


(e) TFA



(f) Google

Figure 3.8: Explicit power law fits to Data. Fit parameters are provided on the plots.



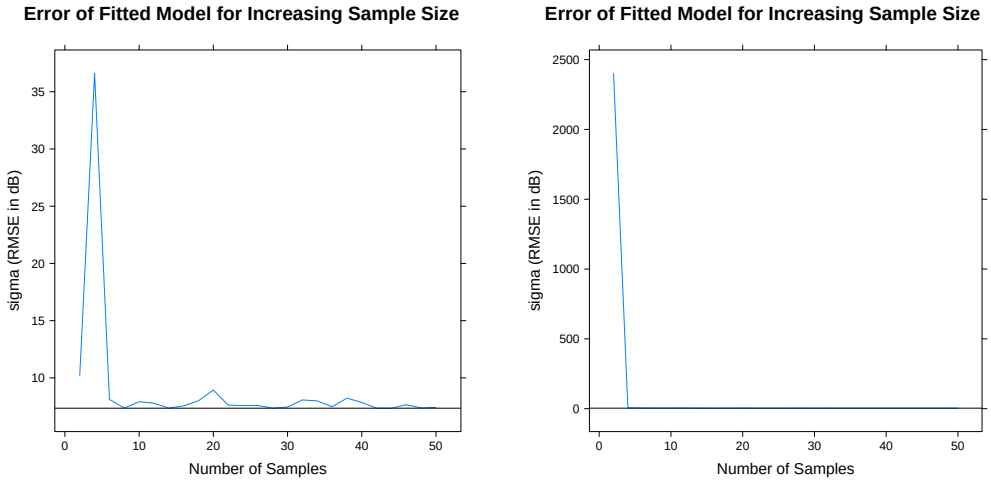
(a) WART

(b) Combined Urban

(c) PDX

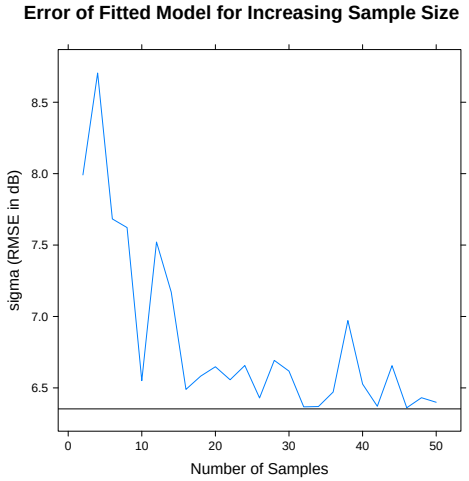
(d) PDX Stumble

Figure 3.9: Explicit power law fits to Data. Fit parameters are provided on the plots.

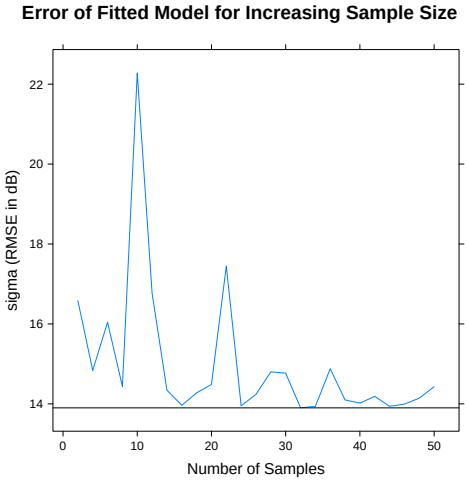


(a) Boulder Downstream

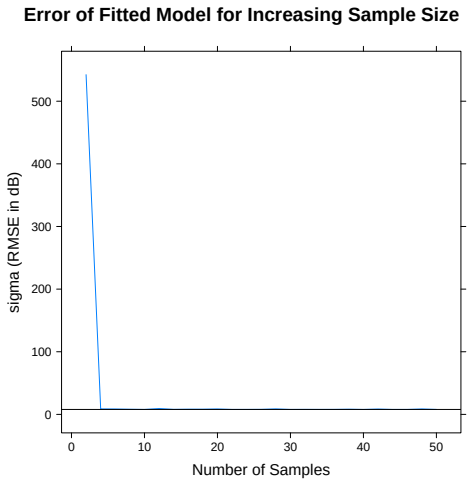
(b) Boulder Upsteam



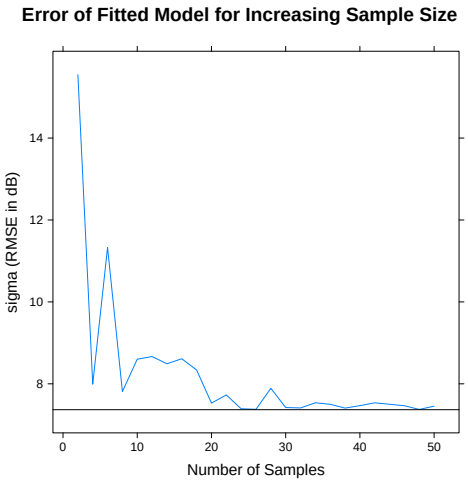
(c) COST-231



(d) WMP

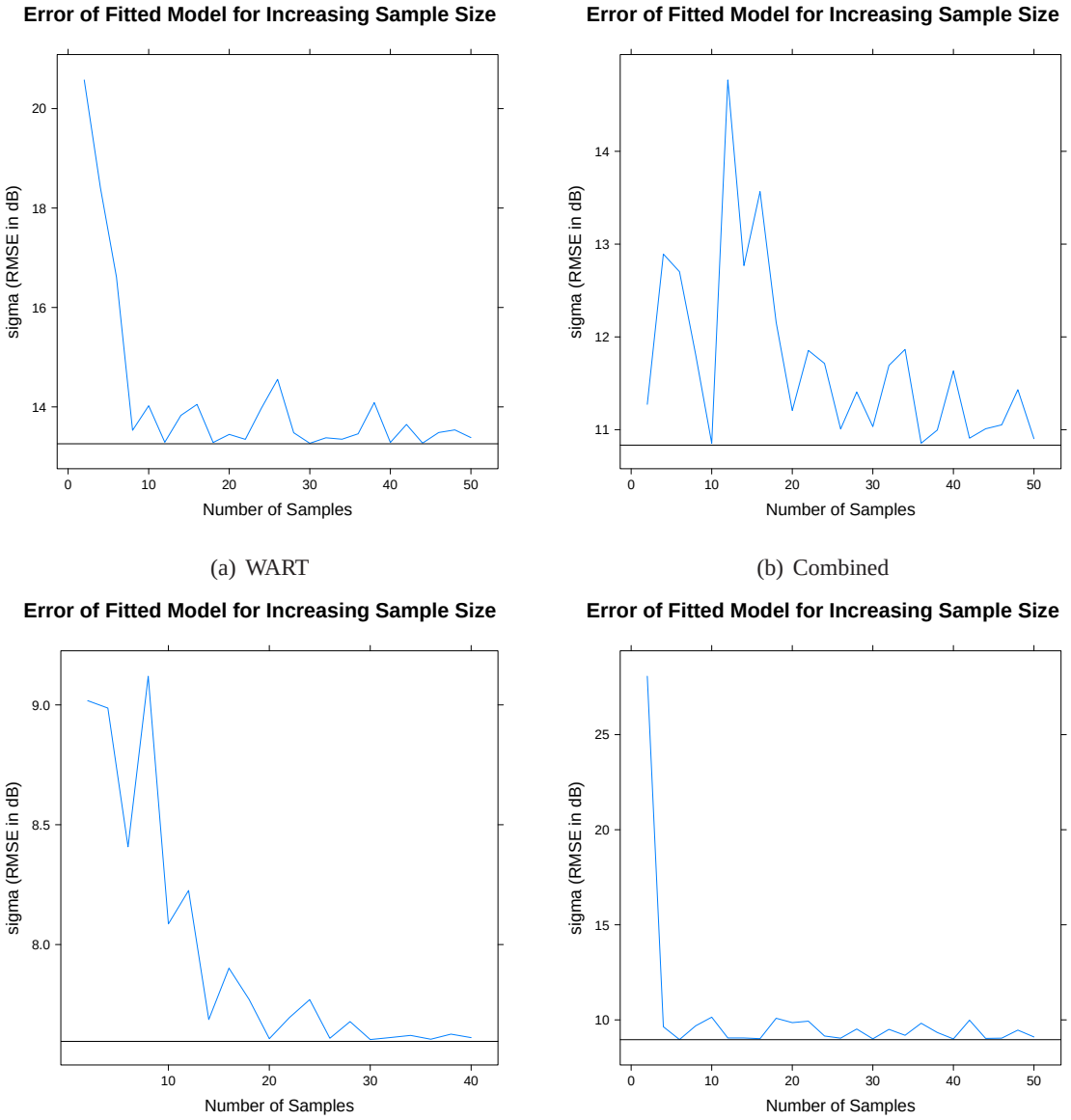


(e) TFA



(f) Google

Figure 3.10: Number of samples required for naïve fit. Plots show fit standard error for fits increasing random samples and a horizontal line is given at the RMSE obtained for all points.



(a) WART

(b) Combined

(c) PDX

(d) PDX Stumble

Figure 3.11: Number of samples required for naïve fit. Plots show fit standard error for fits increasing random samples and a horizontal line is given at the RMSE obtained for all points.

fit in the log/log domain and come up with empirical estimates of the exponent (α) and offset (ϵ):

$$P_r = P_t - (\alpha 10log_{10}(d) + 20log_{10}(f) + 32.45 + \epsilon) \tag{3.3}$$

Figures 3.8 and 3.9 show the resulting fits using this method for each data set and one superset that includes the combination of all urban measurements. One unavoidable side effect of packet-based measurements is that it is impossible to record SNR values for packets that fail to demodulate. Hence, because the 2.4 and 5.8 GHz data is derived from packet-based measurements, low SNR values (and therefore high path loss values) are underrepresented here, which leads to "shallow" fits and unrealistically low values of α. As a result, while it is safe to make comparisons between the 2.4/5.8 GHz data sets, it is not safe to directly compare the slope of the 900 MHz and 2.4/5.8 GHz fits.

Fits are computed using linear least square regression. Table 3.2 lists fitted parameters (α, ϵ) and residual standard error (σ)[2]. Between the 2.4 GHz data sets, there is little consensus about the slope or intercept of this power-law relationship, except that it should be in the neighborhood of $\alpha \approx 2$ and $\epsilon \approx 15$. All fits are noisy, with standard error around 8.68 dB on average for the urban data sets. This residual error tends to be Gaussian, which is also in agreement with previously published measurements (e.g., [183]). However, the size of this error is almost two orders of magnitude from the 3 dB that Rizk *et al.* suggest as an expected repeated measures variance for outdoor urban environments (and hence the expected magnitude of the error due to temporally varying fast fading) [198]. Looking at figure 3.8, it is easy to see that the 2.4 GHz measurements are substantially less well behaved than the COST-231 data, even in comparable environments.

In order to understand how many measurements are needed to create a fit of this sort, successively increasing random samples of the data sets are taken to generate a fit. The rate that residual error of the model (with respect to the complete data set) converges as the subsample size increases can be studied from these results. Figures 3.10 and 3.11 show this plot for each data set. All plots follow a similar trend: the eventual model is closely matched with approximately 20, or at most 40, data points. Table 3.2 gives an approximate minimum sample size for each data set in the column labeled N derived from these plots.

[2] For all intents and purposes, standard error (σ) and RMSE are interchangeable.

Name	α	ϵ	σ	N	Top Three Performing Models by SC-RMSE						Ideal RMSE
wart	1.86	9.05	13.26	15	flatedge	13.73	itu.terrain	13.89	hatao	14.03	1.96
wart/snow	1.92	9.25	13.36	15	itu.terrain	13.93	flatedge	14.16	hatao	14.19	1.87
pdx	2.25	19.53	7.8	5	allsebrook200	8.38	hatal	8.97	davidsons	9.37	1.14
pdx stumble	1.79	27.08	8.96	40	allsebrook400	8.34	itur25	10.50	hatam	10.51	1.02
boulder/ptg	0.79	19.56	7.36	20	allsebrook400	7.90	ecc33m	9.38	hatam	10.47	0.94
boulder/gtp	0.27	10.88	3.67	5	allsebrook400	5.45	hatal.fc	7.15	edwards200	8.51	1.01
cost231	6.25	51.19	6.36	15	edwards200	9.23	hatam	9.99	itur25	10.55	1.23
wmp	0.62	13.74	13.92	15	flatedge	15.34	alsebrook200	16.72	egli	16.83	5.98
tfa	0.95	22.76	7.89	20	herring.atg	8.90	allsebrook200	9.03	flatedge	10.83	1.43
google	0.54	6.15	7.37	30	davidsons	13.56	itu.terrain	16.12	hatal	16.83	2.93

Table 3.2: Summary of results by data set

3.5 Performance Metrics for Path Loss Prediction

The performance of the models is analyzed with respect to several metrics in order of decreasing stringency:

(1) RMSE and Spread-Corrected Root Mean Square Error (SC-RMSE)

(2) Competitive Success

(3) Individual Accuracy Relative to Spread

(4) Skewness

(5) Rank Correlation

3.5.1 RMSE and SC-RMSE

RMSE is the most obvious and straightforward metric for analyzing the error of a predictive model of this sort. As discussed above, for a given model we compute an error value (ϵ as in equation 3.2) for each prediction of each link in each data set. For a given set of links l, in a given data set D and a given model m, the overall RMSE for a given model for a given data set is:

$$RMSE_{m,D} = \sqrt{\frac{\sum_{l \in D} \epsilon_{m,l}^2}{|D|}} \tag{3.4}$$

where $\epsilon_{m,l}$ is the error of model m for link l and $|D|$ is the number of links in the data set D. SC-RMSE is a version of RMSE that subtracts off the expected spread in the measurements from the RMSE. This way, if a given link has large variation in the measurements, then the error a model obtains on that link is reduced by a proportional amount. This has the effect of reducing the error associated with especially noisy links. Figure 3.12 provides an explanatory diagram comparing normal error (ϵ) and spread-corrected error (ϵ'). The spread-corrected error for a given model m and link l is the absolute value of the error, reduced by the standard deviation (σ_l) of measurements on link l:

$$\epsilon'_{m,l} = |\epsilon_{m,l}| - \sigma_l \tag{3.5}$$

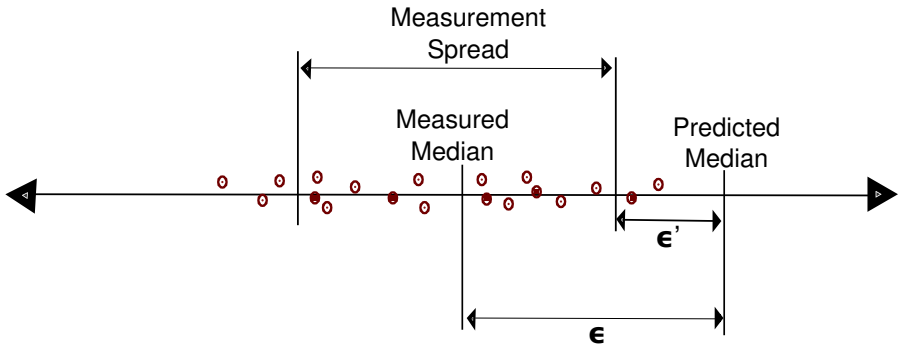


Figure 3.12: Schematic explaining error (ϵ) and spread-corrected error (ϵ') in terms of measurement spread and measured and predicted median values.

Computing SC-RMSE is identical to RMSE as shown in equation 3.4, except ϵ' is substituted for ϵ.

3.5.2 Competitive Success

The competitive success metric is the percentage of links in a given data set that a given model has made the best prediction for. For each link, the model that makes the prediction with the smallest ϵ is recorded. The percentage is computing by counting the number of best predictions for each model and dividing by the total number of links:

$$CS_{m,D} = 100\frac{N_{best,m,D}}{|D|} \tag{3.6}$$

When analyzing many models, if one model (or a set of related models) is dominant for a given environment then it would score near 100 on this metric. Because the percentage points are divided evenly between all models tested, if a large number of models are tested, this metric may become spread too thinly to be useful for analysis (i.e., too many similar models share the winnings and no single model comes out on top).

3.5.3 Individual Accuracy Relative to Spread

The individual accuracy metric is the percentage of links where the given model is able to make a prediction within one or two standard deviations of the measured spread:

$$IA_{m,D} = 100\frac{\sum_{l \in D} \begin{cases} 1 & |\epsilon m, l| < k\sigma_l \\ 0 & o.w. \end{cases}}{|D|}; k = 1, 2, ... \tag{3.7}$$

where k is how many standard deviations to use for the metric. In the following analysis, results for $k = 1$ and $k = 2$ are used.

3.5.4 Skewness

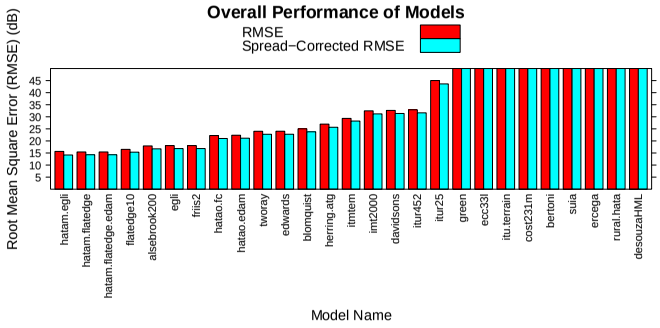
The fourth metric is skewness, which is simply the sum of model error across all links:

$$S_{m,D} = \sum_{l \in D} \epsilon_{m,l} \tag{3.8}$$

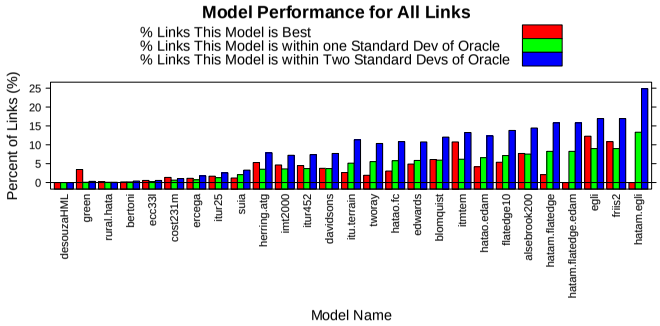
This metric highlights those models that systematically over- and underpredict. Some applications may have a particular cost/benefit for under or overpredictions. Models that systematically overpredict path loss (and therefore underpredict received signal strength) score a high value on this metric. Models that systematically underpredict score a large negative value. Models that make an equal amount of under- and overpredictions will score a value of zero.

3.5.5 Rank Correlation

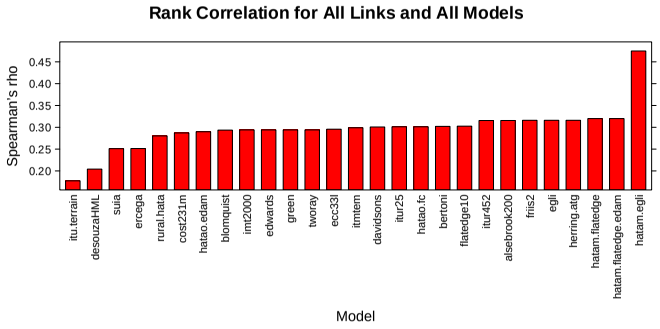
The final metric is rank correlation using Spearman's ρ[3] . In some applications, predicting an accurate median path loss value might not be necessary so long as a model is able to put links in a correct order from best to worst (consider, for instance, the application of dynamic routing). Spearman's ρ is a nonparametric measure of statistical dependence and in this application describes the relationship between ranked predictions and oracle values using a value between -1.0 (strong negative correlation) and 1.0 (strong positive correlation).



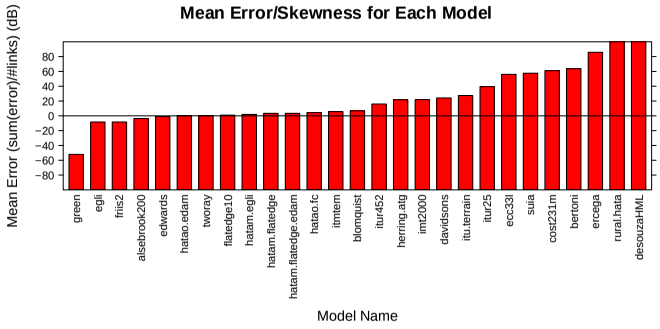
(a) RMSE



(b) Success/Accuracy

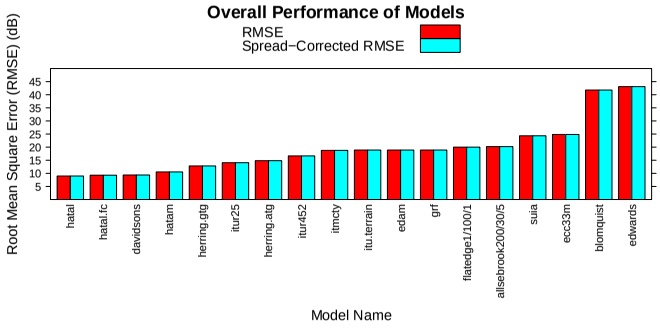


(c) Rank Correlation

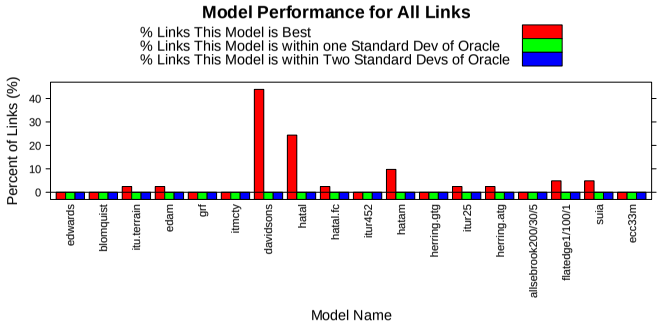


(d) Skewness

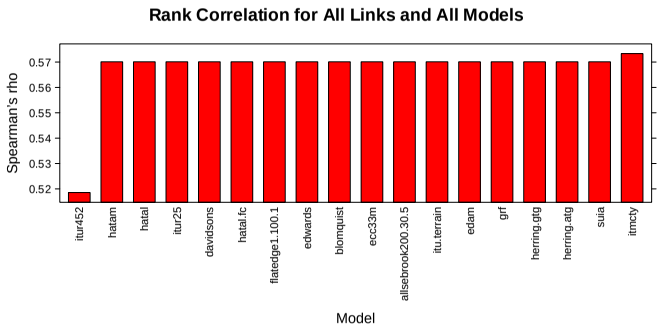
Figure 3.13: Five metric results for **WMP** data set



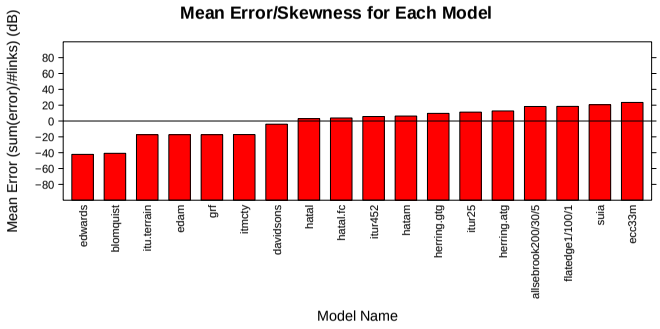
(a) RMSE



(b) Success/Accuracy

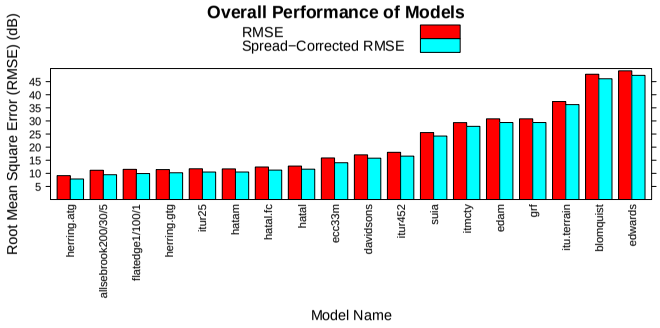


(c) Rank Correlation

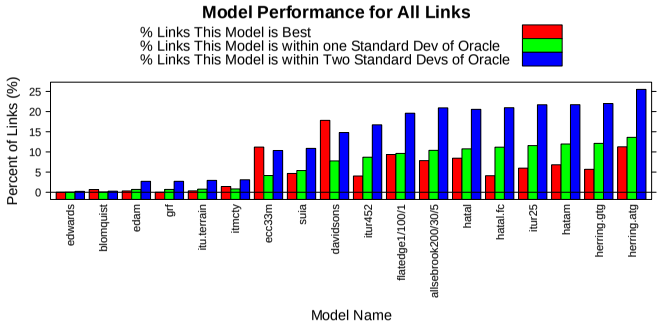


(d) Skewness

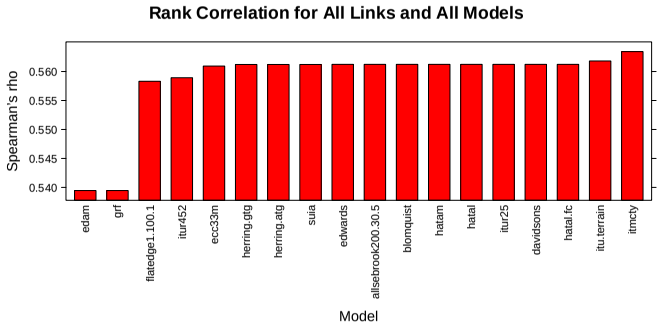
Figure 3.14: Five metric results for **PDX** data set



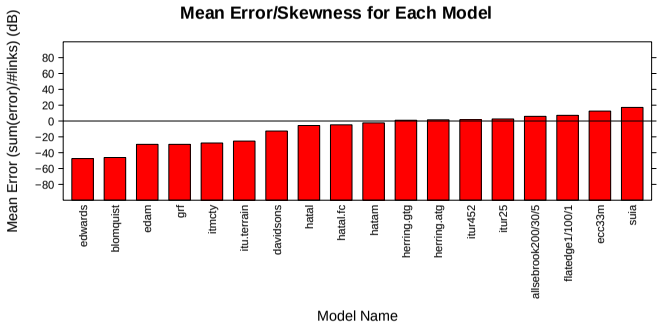
(a) RMSE



(b) Success/Accuracy

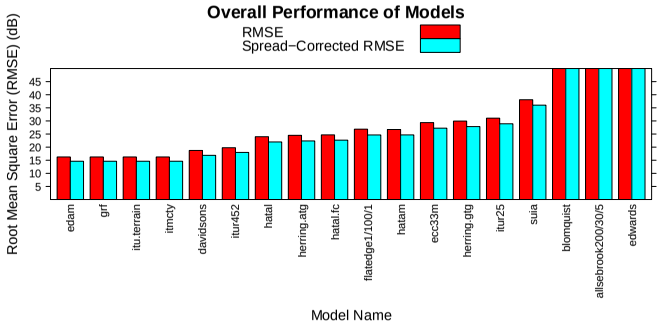


(c) Rank Correlation

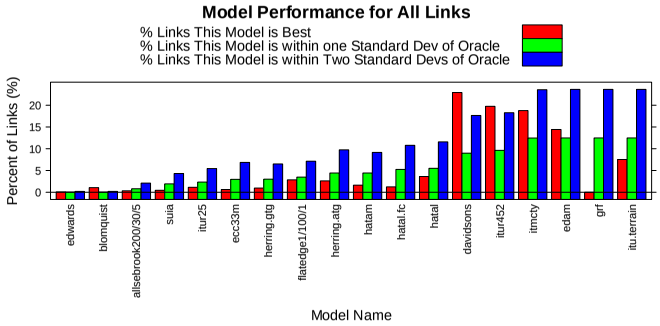


(d) Skewness

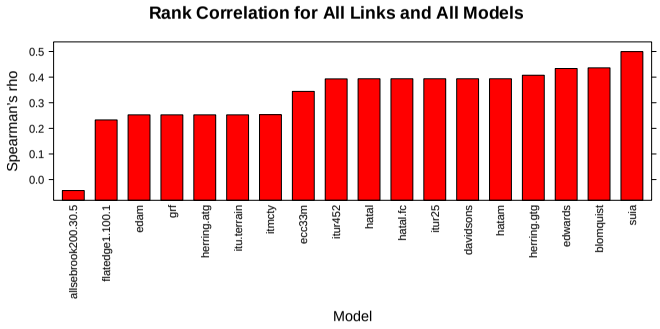
Figure 3.15: Five metric results for **PDX/stumble** data set



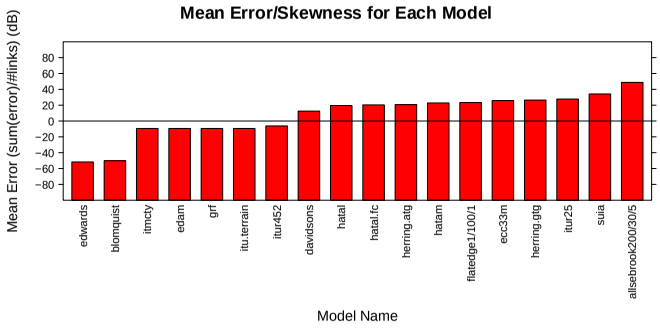
(a) RMSE



(b) Success/Accuracy

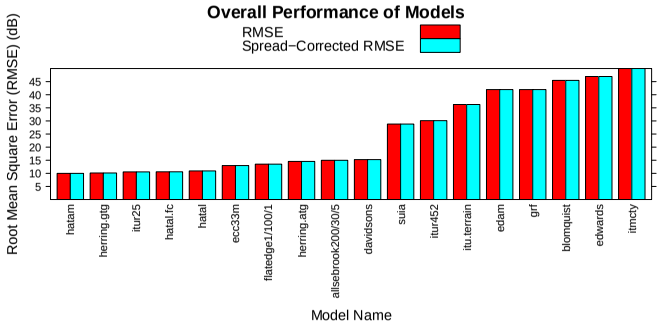


(c) Rank Correlation

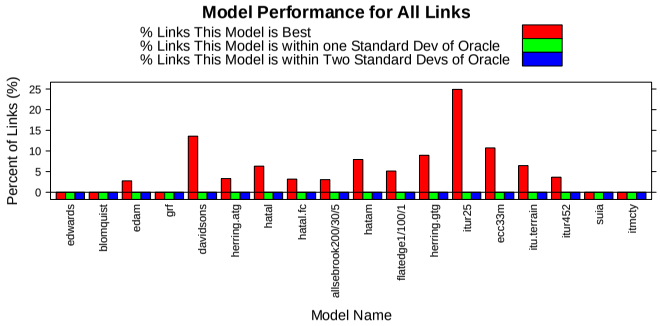


(d) Skewness

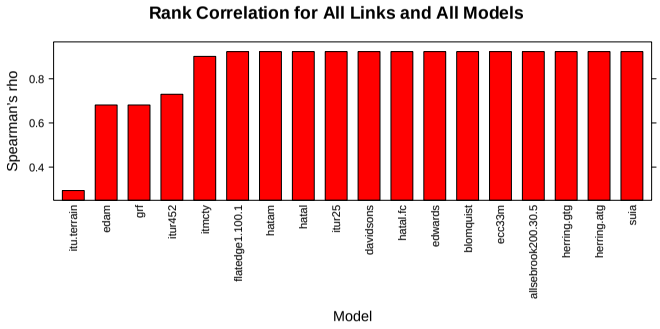
Figure 3.16: Five metric results for **WART** data set



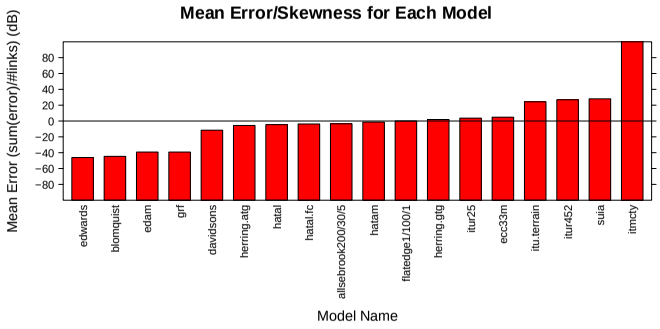
(a) RMSE



(b) Success/Accuracy

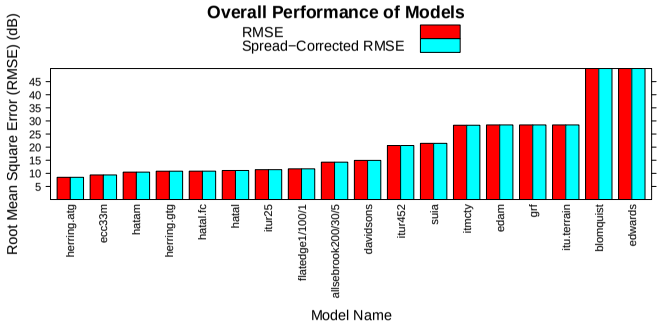


(c) Rank Correlation

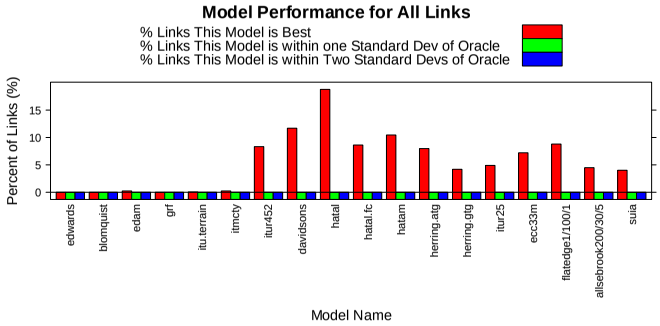


(d) Skewness

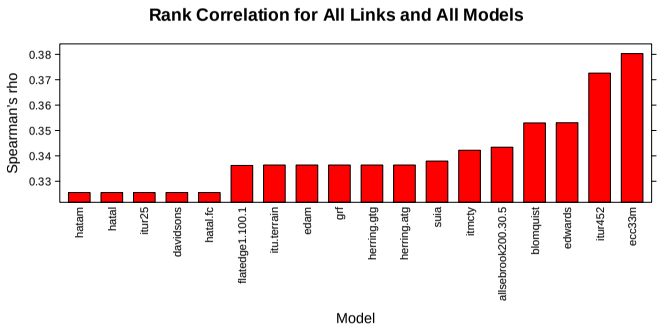
Figure 3.17: Five metric results for **COST231** data set



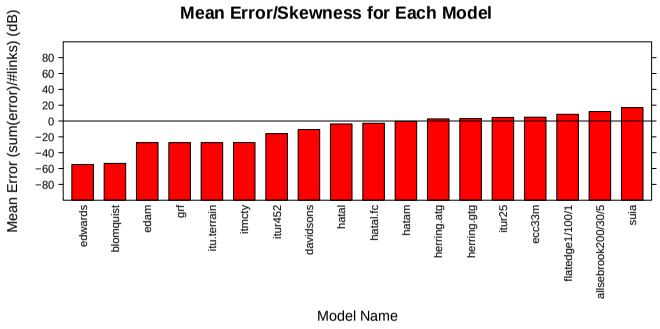
(a) RMSE



(b) Success/Accuracy

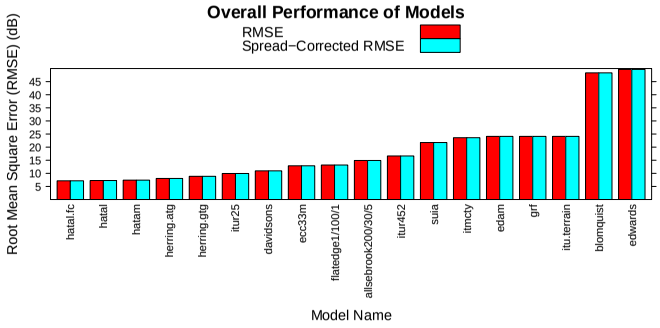


(c) Rank Correlation

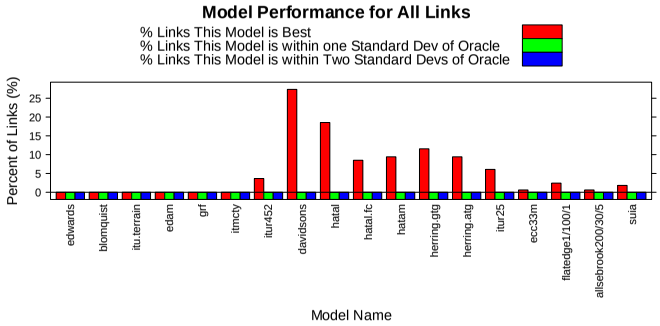


(d) Skewness

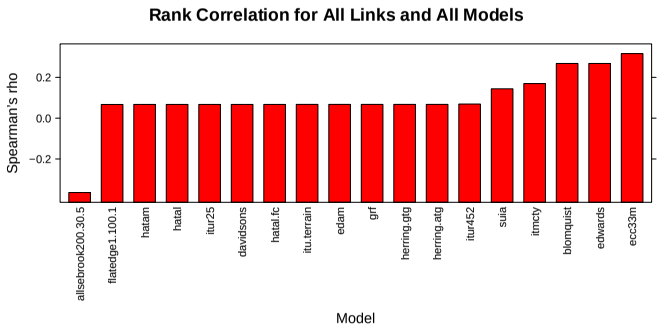
Figure 3.18: Five metric results for **Boulder/ptg** data set



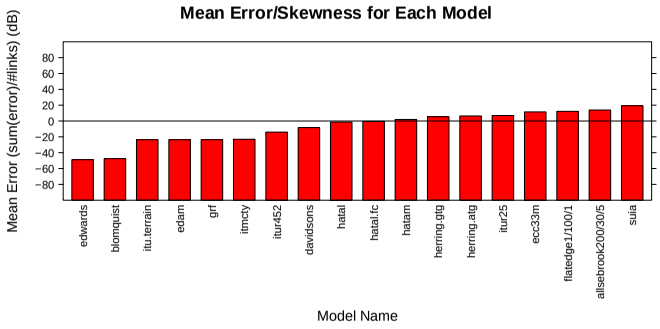
(a) RMSE



(b) Success/Accuracy

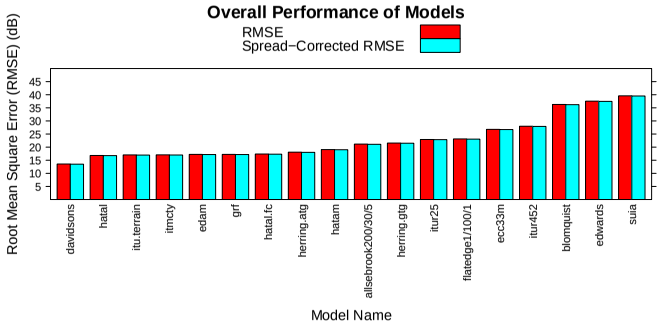


(c) Rank Correlation

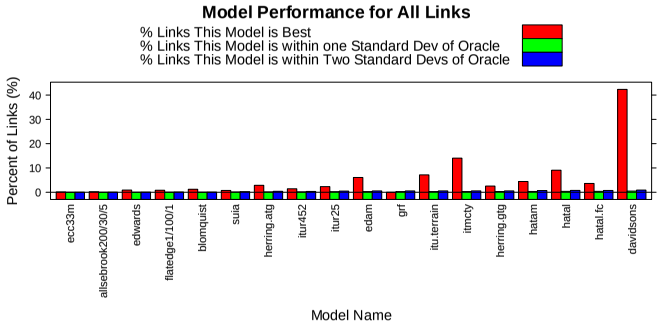


(d) Skewness

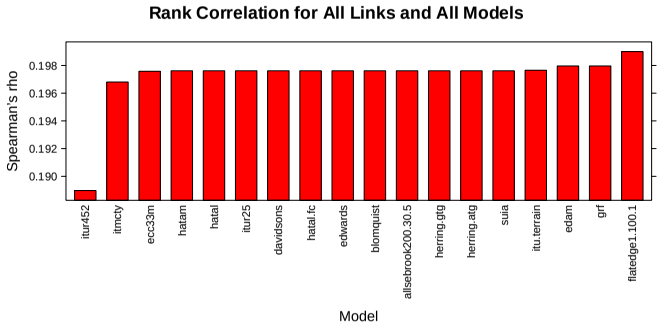
Figure 3.19: Five metric results for **Boulder/gtp** data set



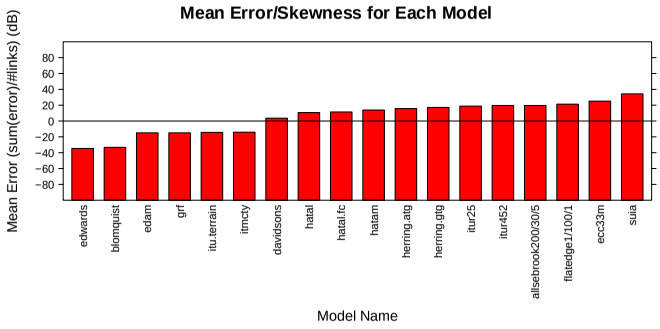
(a) RMSE



(b) Success/Accuracy

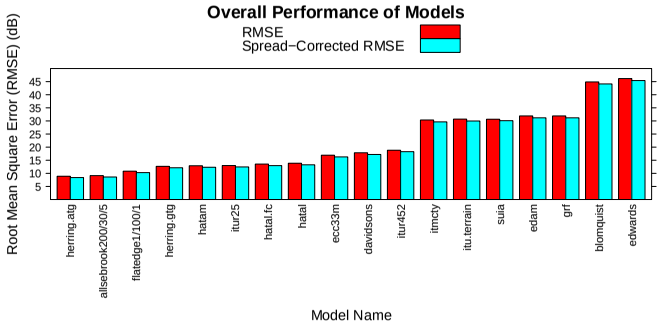


(c) Rank Correlation

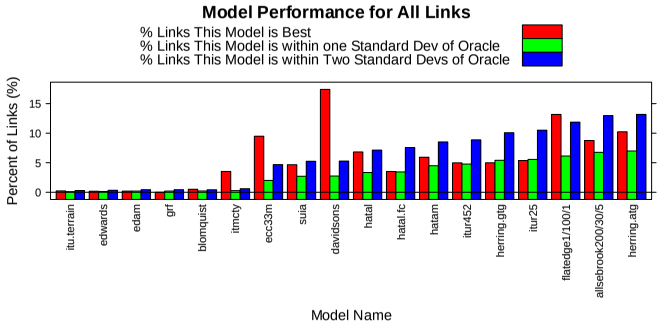


(d) Skewness

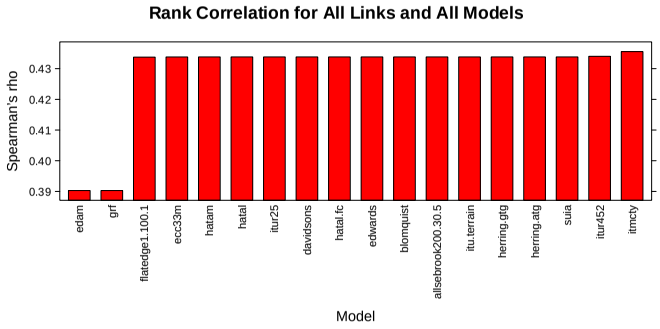
Figure 3.20: Five metric results for **Google** data set



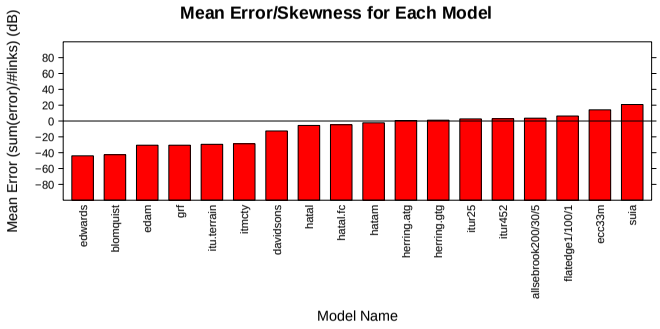
(a) RMSE



(b) Success/Accuracy

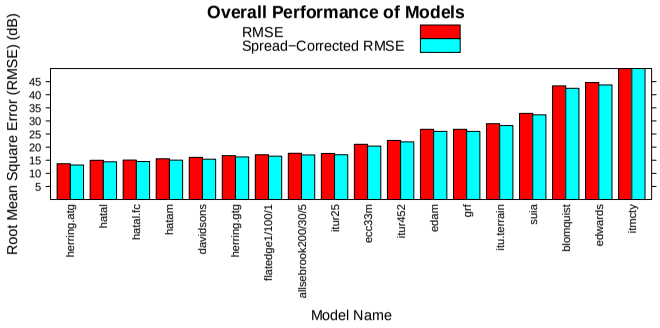


(c) Rank Correlation

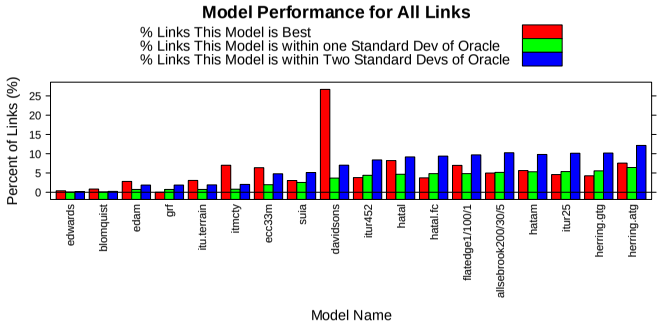


(d) Skewness

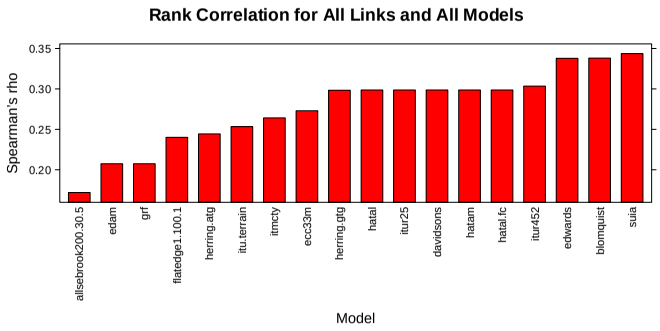
Figure 3.21: Five metric results for **TFA** data set



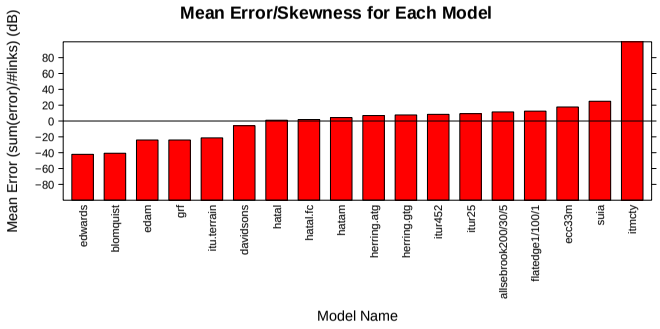
(a) RMSE



(b) Success/Accuracy

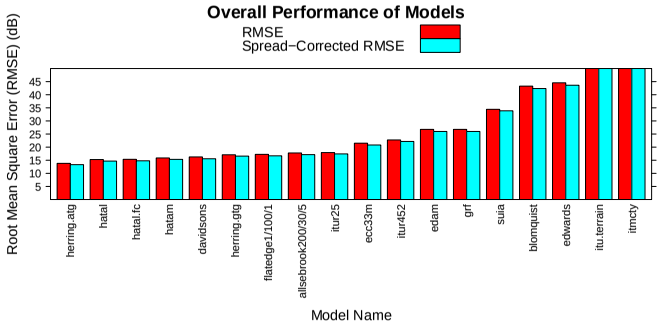


(c) Rank Correlation

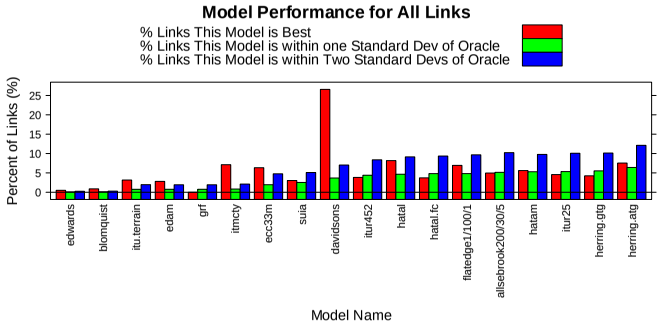


(d) Skewness

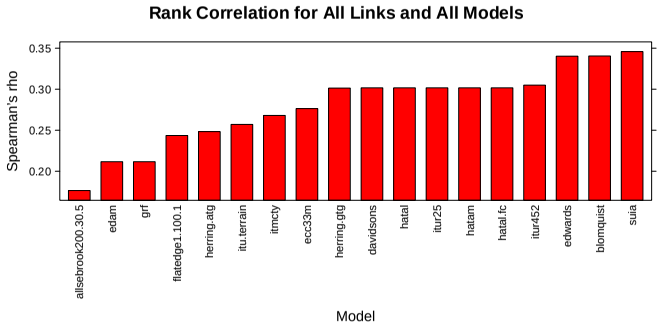
Figure 3.22: Five metric results for all **Urban** data sets



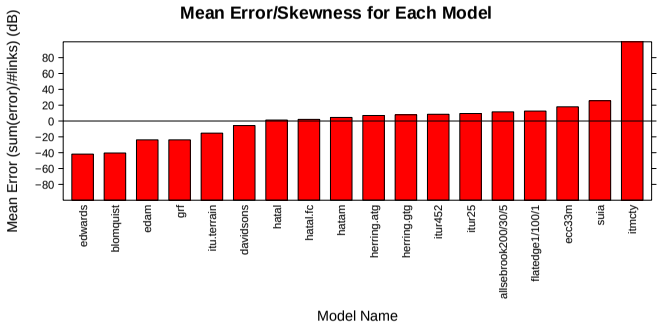
(a) RMSE



(b) Success/Accuracy



(c) Rank Correlation



(d) Skewness

Figure 3.23: Five metric results for all data sets combined

3.6 Results

Figures 3.13-3.23 show the results of these metrics for each data set and all (urban) data sets combined. To simplify the plots, only results from the 18 best-performing models (30 for the rural data) are included. Because the urban and rural data sets were best modeled by different algorithms, a slightly different set of models is shown for each of these. However, the urban data sets present results from the same subset of models so that results are easily comparable.

Looking first at the results for the rural (WMP) data, the best-performing models achieve an RMSE on the order of 15 dB. The best models are the Alsebrook model (with its terrain roughness parameter set to 200m) at just under 18 dB RMSE (16.7 dB when corrected), and the Flat-Edge model (with 10 "buildings" presumed) at 16.5 dB RMSE (15.3 dB when corrected). In the urban data sets, the urban models do much better in terms of RMSE. The best models achieve an RMSE on the order of 10 dB, and the worst (of the best) approach more than 50 dB. The overall winners are the Hata model, the Allsebrook-Parsons model, the Flat-Edge model, and the ITU-R model. This follows from expectations because all of these models were derived for predicting path loss in urban environments. The Hata model and Allsebrook-Parsons models are based on measurements from Japanese and British cities respectively. The Flat-Edge model is a purely theoretical model based on the Walfisch-Bertoni model, which computes loss due to diffraction over a set of uniform screens (simulating buildings separated by streets). Table 3.2 provides the top three models by SC-RMSE for each data set and their corresponding error.

The second metric, competitive success, is shown with the leftmost (red) bar in the second of each set of plots. For most of the data sets, there is no clear winner, with the best models sharing between 10 and 15 percent of the winnings. This indicates that there is no single model that outperforms all others. However, there are a few exceptions. For the PDX data set, the Davidsons model takes 40% of the winnings. In the COST-231 data set, the ITU-R 25 model takes 30%. In the Google data set, the Davidson's model takes more then 30%. And, in the downstream Boulder measurements (boulder/gtp), the Davidon's model again takes 25% of the winnings. There are not, however, one or two models that outperform all others in a large subset

[3] Kendall's τ would be an equally appropriate metric, but is slower to compute.

of the data. Hence, we can conclude that the choice of the most-winning model is environment-dependent.

The third metric is percentage of predictions within one (or two) standard deviations of the true median value. This metric requires multiple measurements at each point in order to estimate temporal variation in the channel. Of the data sets, six have this data available: WMP, COST-231, PDX/Stumble, Google, TFA, and WART. For the WMP data the best-performing models (Allsebrook-Parsons, Flat Edge, Herring Air-to-Ground, and ITU-R) score between 10% (for within one standard deviation) and 20% (for within two standard deviations) on this metric. Similar results can be seen for our other data sets, but with different winners. For the PDX/Stumble data, the winners are Herring Air-to-Ground, Hata, and ITU-R 25. For the WART data set, the winners are the ITM, ITU-Terrain, and Blomquist. For the COST-231 data set, the winners are Herring Air-to-Ground, Hata, and Allsebrook-Parsons. Again, the best-performing model appears to be largely environment-dependent.

The fourth metric is skewness. The interpretation of this metric is largely application-dependent, i.e., it is hard to know in advance whether over- or underestimates are more harmful. If a model makes an equal amount of over- or underestimates (resulting in zero skewness), but has a large RMSE, is it better than a model that systematically overestimates but has a small RMSE? The Hata model is particularly well behaved by this metric, producing a value near zero for all data sets. As one would expect, the Hata-derived models perform similarly (i.e., ITU-R 25, Davidsons, etc.). The rest of the models seem to vary largely from data set to data set, although ITU-R 452 performs well for some data sets.

The final metric is rank correlation. For just about all of the models a rank correlation around 0.5 is observed, which indicates a moderate (but not strong) correlation between measured and predicted rank orderings. Models that perform particularly poorly by this metric achieve values much lower on occasion. A result near zero indicates that there is no noticeable correlation between rank orderings. The COST-231 rank correlations are substantially higher than all other data sets. This may be related to the fact that the COST-231 data more closely fits theoretical expectations of the relationship of path loss to distance. Hence, models that use something like Friis equation at their core will produce rank values that are closer to data in this data set. Overall, however, there does not seem to be a consensus about which model performs best at rank ordering—the winners are different for each data set.

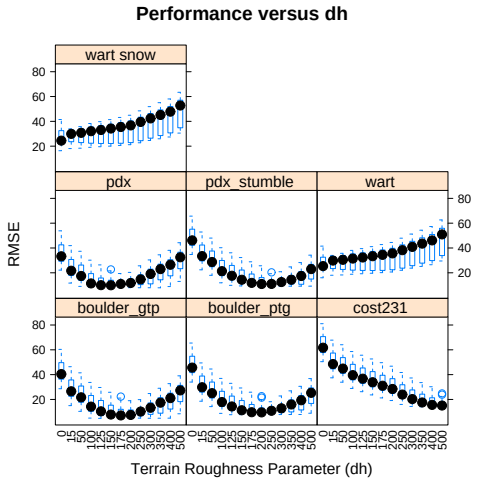
3.6.1 Explicit Parameter Fitting

In order to get an idea of minimum obtainable error with these models, two well-performing models that have tunable parameters are used: Allsebrook-Parsons and Flat-Edge. The experiment proceeds by searching the parameter space to find the best-possible parameter configuration for each[4]. The Allsebrook-Parsons model takes three parameters (besides carrier frequency, which is common to nearly all the models): Δh, a terrain roughness parameter (in m), h_0, the average height of buildings (in m), and d_2, the average width of streets (in m). The Flat-Edge model also takes three parameters: n, the number of buildings between the transmitter and receiver, h_0, the average height of these buildings (in m), and w, the street width (in m). After sweeping the parameter space, an Analysis of Variance (ANOVA) is used to determine the parameters that best explain the variance in the data.

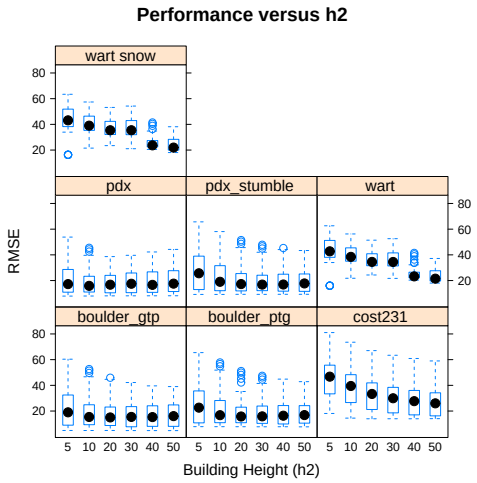
For the Allsebrook-Parsons model, the Δh and h_2 parameters are both important. For the Flat-Edge model, h_0 is the only significant parameter. Figure 3.24 shows the response (in terms of RMSE) for tuning these parameters. The optimal values can be determined from the minima of these plots and a similar approach could be carried out with any subset of the data. However, *the optimal parameters for one datum are not usually in agreement with others, forcing a compromise in terms of accuracy and specificity*. Even with cherry-picked parameters, the RMSE is still in the neighborhood of 9-12 dB, which is too large for most applications.

If 9 dB is considered to be the minimum achievable error of a *well-tuned* model, it is interesting to note that approximately the same performance can be achieved with a straight line fit through a small number (≈ 20) of measurements as was shown in section 3.4. In [73], the authors found similar bounds on error (6-10 dB) attempting to fit a single model to substantial measurement data at 1900 MHz. If the domain of interest is network planning, and it is not possible to make measurements of a network (because it does not yet exist), then tuning an *a priori* model may be the right approach to take. However, if the goal is modeling the path loss of a network that can be directly studied, and taking 20 (randomly distributed) measurements is reasonably cheap, then this approach seems easy to advocate by comparison.

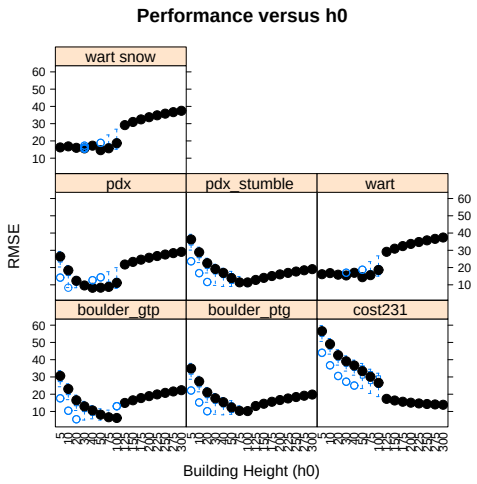
[4] Data from the Boulder, WART, and PDX data sets were used for this experiment.



(a)



(b)



(c)

Figure 3.24: Explicit parameter fitting for the Allsebrook-Parsons and Flat-Edge model parameters.

3.6.2 Factors Correlated with Error

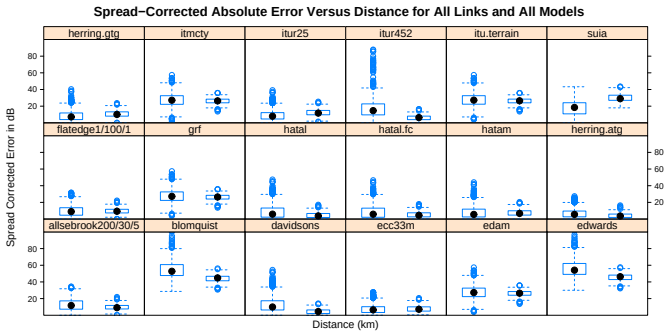
Overall, these results are not terribly impressive. Even in the mean case, the best models with their best parameter settings cannot achieve an error of less than 15 dB for the rural measurements and approximately 9 dB for the urban data sets—three to five orders of magnitude from the correct value. More permissive performance metrics show the models are unable to widely succeed at seemingly simple tasks of rank-ordering links, or making predictions within two standard deviations of the measured value. This begs the question: is there some common source of error that is affecting all models?

In order to understand which variables may serve to explain model error, a factorial ANOVA was performed using spread-corrected error as the fitted value and transmitter height, receiver height, distance, line-of-sight (a boolean value based on path elevation profile), and data set. Although all of these variables show moderate correlations (which speaks to the fact that many models add corrections based on these variables), some are much better explanations of variance than others. Perhaps not surprisingly, distance and data set name are the biggest winners with extremely large F-values[5](16,687.34 and 52,375.54, respectively, and 14,156.54 when combined). Figures 3.25-3.27 plot the relationship between error and link distance for each of the best-performing models—the relationship is plain to see. This leads to the conclusion that the best results can be obtained when an appropriate model is known for a given environment, and when the model is designed for the same distances of links being modeled. *Using models outside of their best environment and best distance coverage will result in substantial error*. This conclusion motivates hybridized models that change their approach based on the environment or length of links being modeled.

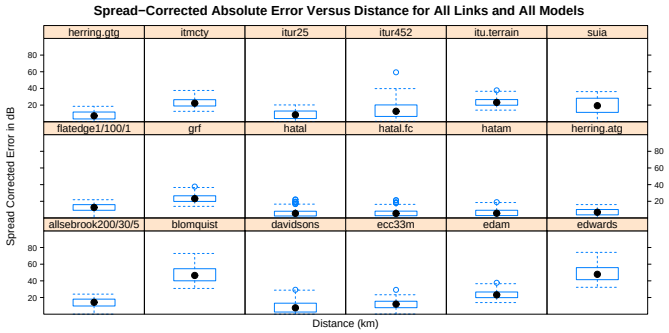
3.6.3 Distance-Hybrid Models

To understand the possible benefit of hybridized models, three hybrid models were implemented and applied to the WMP data. The WMP data was chosen because it includes the largest variety of link lengths. The first uses the Hata model (for medium cities) for links under 500m (where it is well-performing) and the Flat-Edge model (with 10 “buildings”) for longer links (hatam.flatedge10). This new model performs

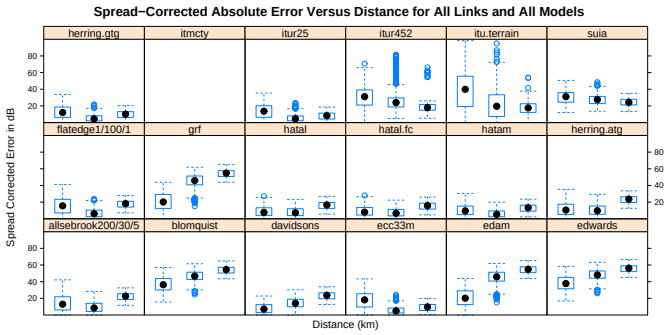
[5] The F-value is a statistic that describes the ratio between explained variance and unexplained variance. Or, put differently, the ratio of between-group variability to within-group variability.



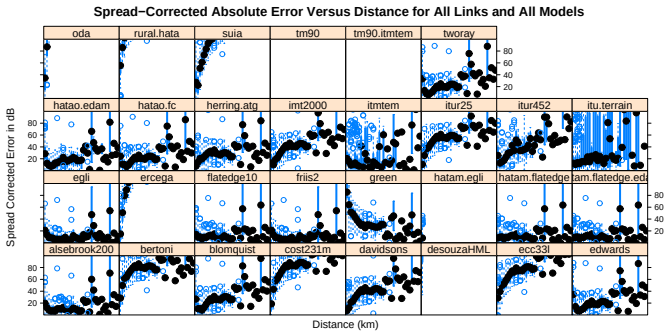
(a) Boulder Downstream



(b) Boulder Upsteam

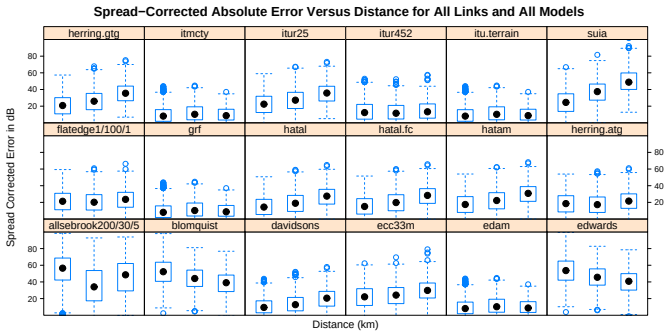


(c) COST-231

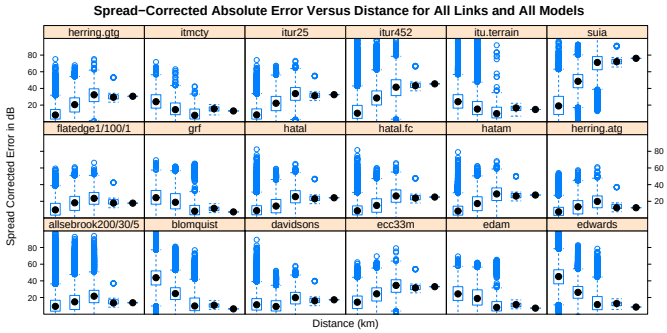


(d) WMP

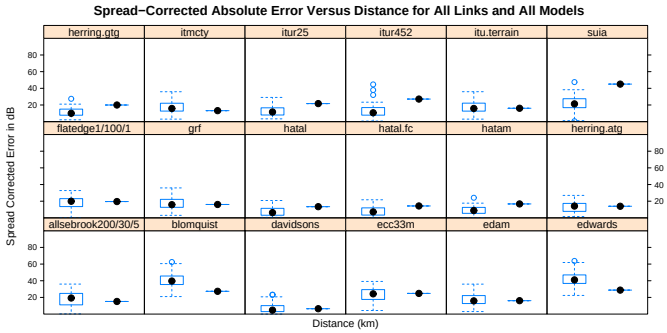
Figure 3.25: Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer.



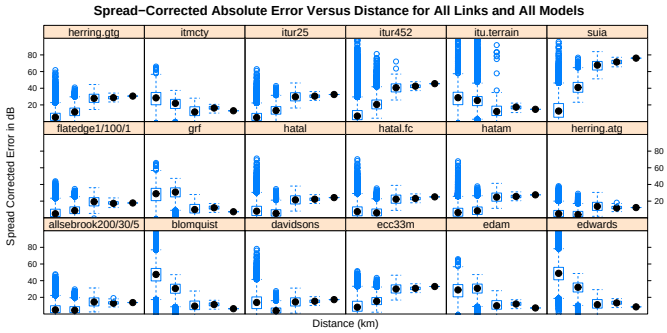
(a) WART



(b) Combined

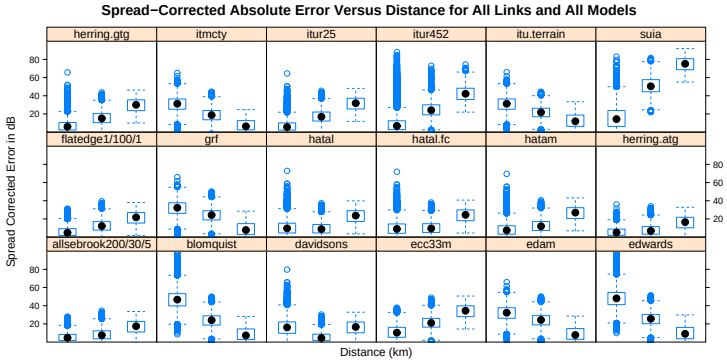


(c) PDX

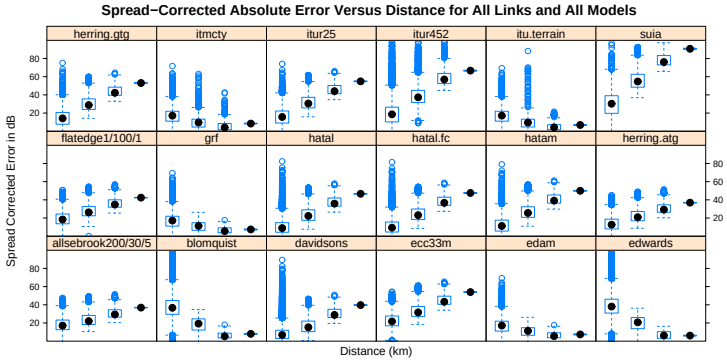


(d) PDX Stumble

Figure 3.26: Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer.



(a) TFA



(b) Google

Figure 3.27: Correlation between model accuracy and link distance for each data set. Distance is bucketed by kilometer.

marginally better than all other models, producing a corrected RMSE of 14.3 dB. Very slightly better performance is achieved by combining the Hata model with the Egli Model (14.2 dB RMSE). The third combination uses the TM90 model for links less than 10 miles and the ITM for longer links (tm90.itmtem). However, this combination is not well-performing with respect to the measurements[6]. Treating this tuning and hybridization as an optimization problem with the goal of producing the best-performing configuration of existing protocols is a project for future work. Taking this approach however, one must be careful to avoid overfitting a model to the data available.

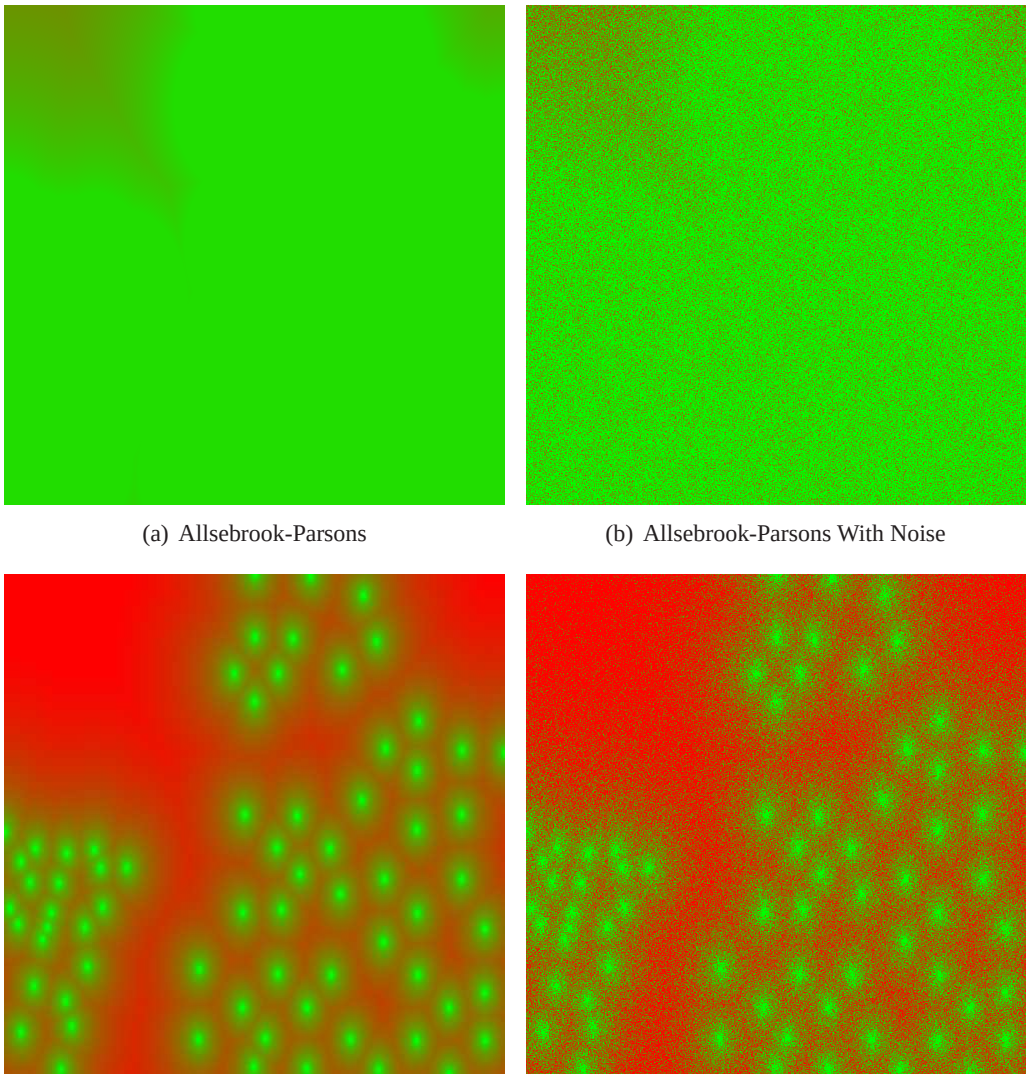
3.6.4 Practical Interpretation

As an example of what these performance results mean for real applications, consider figure 3.28, which shows a predicted coverage map for the Portland MetroFi network using two well-performing models *tuned to their best-performing configurations*. Maps with zero-mean 12 dB Gaussian noise, which approximates the expected residual error from these models, have also been included. To generate these maps, the 2 km by 2 km coverage area was divided into a 500x500 raster and each pixel is colored based on predicted received signal strength, linearly interpolated between red (at -95 dBm) and green (at -30 dBm). For each pixel, the predicted path loss from all 72 APs is computed and the maximum value is used to color the pixel.

Comparing these maps to the empirical and operator-assumed coverage maps in figure 4.1, it is clear to see that there is no consensus on what the propagation environment looks like. The Hata model may produce the picture that is closest to the measurements, but the results show that it is not the best-performing model overall. Moreover, the Allsebrook-Parsons model, which is well-performing overall, and has been tuned to its best configuration, produces a map that is in stark disagreement with reality.

Yet, the future holds promise. Consider the final column in Table 3.2, which gives the RMSE for each data set if we choose to take only the best prediction among all of the predictions made by the 30 models and their configurations. This represents one version of a minimal achievable error in a world with a perfectly hybridized model that always knows which model to use when. In this scenario, we can see a very

[6] This approach is of special interest because it is the one advocated by the FCC in recent discussions about whitespaces transmissions in 3 GHz and below 900 MHz bands. In [141] in particular, the FCC suggests the use of the ITM for long distance predictions and the TM90 model for shorter (less than 10 mile) predictions.



(a) Allsebrook-Parsons (b) Allsebrook-Parsons With Noise

(c) Hata (d) Hata with Noise

Figure 3.28: Comparison of predicted coverage maps for Portland, Oregon using two well-performing models, with and without same scale Gaussian error included. True green indicates predicted recieved signal at -30 dBm and true red indicates predicted received signal the noise floor (-95 dBm). Intermediary values are linearly interpolated between these two color values.

attractive bound on error—as low as 1 dB. This indicates that there is still room for improvement. If we were able to determine the situations when each model is likely to succeed, then it is reasonable to assume that it is possible to construct a single hybrid model that is more accurate than the sum of its parts. This thesis takes the perspective that an approach that marries appropriate (possibly hybridized) modeling techniques with directed measurements, will result in a better complete system than can be accomplished with either

measurement or model-tuning alone.

3.6.5 Miscellaneous Observations

This section discusses a few important miscellaneous observations based on the results above.

3.6.5.1 Modeling Directional Antennas is Challenging

One interesting additional observation from this data is that modeling path loss from directional transmitters is especially difficult. This can be seen in the fact that the data from the directional CU-WART testbed is particularly noisy. There has been at least one attempt to model this phenomenon explicitly in the past [85], but even using this correction, the error in prediction of directional propagation is still much greater than for omnidirectional transmitters. To this end an empirical supplementary model was derived from an extensive set of measurements. This model is called the EDAM and is described in detail in appendix A. Although this model is not particularly winning in the analysis here, in prior work it was shown to be better than simple models found in common simulators in at least one application [30]. While not a complete solution, EDAM is a solid first step in the direction of an appropriate modeling strategy for antenna directivity.

3.6.5.2 Models that Generate Errors

It is worth noting that some algorithms will generate error conditions when used outside of their intended coverage. If these models are given the benefit of the doubt and only used where no errors or warnings were generated, the overall performance looks better. For instance, the corrected RMSE for ITM (with parameters for a temperate environment) on the WMP data set improves from 28.2 dB to to 23.1 dB if the most eggregious errors are discarded (which stem from problems predicting refraction over terrain for certain terrain types, and is only 290 of 2,492 predictions) and down to 17.3 dB when only those predictions that generate zero warnings are used (which usually stem from links that are too short and are only 696 of 2,492 predictions). This is a substantial improvement—at 17.3 dB corrected RMSE, the ITM is performing on par with the best of the other models.

3.6.5.3 Prediction in Rural Environments is Challenging

In a result that appears completely counterintuitive, the rural data set is much more difficult to model than the urban data sets. To look for sources of systematic error, covariance (correlation) between "best prediction error" (the error of the best prediction from all models) and various possible factors was analyzed. There appears to be no significant correlation between carrier frequency (and therefore neither modulation scheme nor protocol) or antenna geometry. However, there is a large correlation between error and distance. It is hypothesized that the reason the WMP data is especially difficult to model may have to do with two factors: (1) Because researchers have assumed that rural environments are "easy" or "solved", there has been substantially more work in developing (empirical) models for urban environments. The majority of state-of-the-art Rural models, on the other hand, are largely analytical and were mostly developed 30 or more years ago (i.e., the ITM) (2) This data set has an exceptionally large variety of link lengths, and as has been shown, prediction error is strongly correlated with distance for many models. However, more work is needed to confirm or deny these hypotheses.

3.7 Evaluation of Raytracing Systems

Ray-tracing (or many-ray) models, which compute the interactions between many rays and obstacles using the UTD or Finite Difference Method (FDM), are considered by many to be the state-of-the-art in path loss prediction. These models differ from the comparatively simple models discussed so far in that they consider the combined effect of constructive and destructive interference along many competing paths. These models have been widely integrated into commercial wireless planning software (e.g., [51, 187, 239]). Becuase of the large licensing cost of this software and significant data requirements (building models are required for outdoor prediction, and often architectural floorplans are required for indoor prediction), their use is generally excluded from all but the most demanding (and well-funded) applications. Indeed, individuals who design wireless planning tools often find that while ray-tracing methods are the highest powered models in their software, they are typically used seldomly as compared to more simplistic (often probabilistic empirical and data corrected) models such as those investigated in this section [238].

In order to understand how well ray-tracing solutions to wireless planning work, trial licenses from two well-regarded software vendors were obtained: EDX Wireless [239], and REMCOM [187]. The aim here is to predict the path loss at points on the CU campus and compare them to measurements from a set of fixed WiMax BSs at those same points. Chapter 5 describes this ground-truth data and how it was collected. This data set was chosen because of the availability of building vector data on the CU campus, while similar data was not available for the environments in which the other measurements were collected. Because ray-tracing software is clearly very dependent on the environmental data used to make predictions, three environmental data sets of increasing fidelity are used:

- Buildings as Rectilinear Shapes: in this data set, each building on the CU campus is modeled as a single polygon of an approximately correct height and footprint. This data set was manually created by EDX engineers and hence is stored in a proprietary format, which prevents use with the other (REMCOM) software. This dataset is typical of what a customer would use in planning a network [238].

- Crowdsourced Building Vector Data: for this dataset, building data was extracted from the Google Earth 3D warehouse [209]. Because the Google Sketchup software was developed in Boulder, Colorado, the 3D warehouse data for the CU campus is particularly good as it was designed by the Sketchup company itself for internal testing. This data is available through the 3D warehouse website and can be downloaded as a set of several hundred Collada files [102]. With some care, this Collada format is converted to the Stereo Lithography (STL) format, which is more widely useful. The STL format descibes the building extents as a set of positioned facets (2D polygons) and their normal vectors. As part of the REMCOM Wireless Insight API [189], this STL format can be converted to a standard Shapefile, which describes buildings with polygonal shapes. The datset constitutes a high level of fidelity which has been obtained via a large amount of work by many individuals, yet its accuracy has not been independently verified.

- Light Detection and Ranging (LiDaR) Data: High resolution LiDaR data was obtained from geography researcher Shane Grigsby at CU, who collected the data in collaboration with the CU En-

vironmental Center [155] and National Science Foundation (NSF) National Center for Airbourne Laser Mapping (NCALM) [74]. This dataset contains more than 200 million points which describe the height (and to some extent the "hardness") of all obstacles on the CU campus in very high resolution. The dataset is described in detail at [87]. This data set constitutes the highest resolution data obtainable for an outdoor environment, and is very costly to collect.

After some discussion with engineers at EDX and REMCOM [238, 188], it became clear that state-of-the-art wireless planning software is simply incapable of working with data at the fidelity offered by the LiDaR data set. Indeed, converting such a point-data set to a raster data set is a complicated task, and coverting a raster data set to a vector format is an open research question. The middle data set, derived from crowd-sourced data, was able to be used in the REMCOM software after some work to reformat it. However, this data set is far too complex for use in the EDX software [238]. Hence, results are only available for the REMCOM software using the second data set and the EDX software, using the most simplistic data. Although this substantially hinders the power of the results here, it is not feasible at present to perform a factorial analysis of the sensitivity of ray-tracing results to the fidelity of input data. Instead, this question is left for future work.

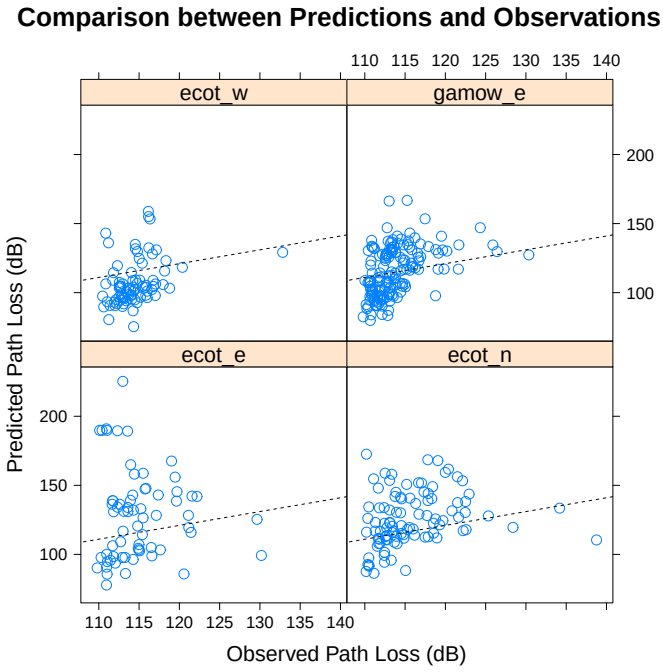
3.7.1 Case Study: REMCOM Wireless Insight and Crowd Sourced Building Models

For this scenario, the WiMax measurements for the five BSs described in section 6.1 are compared to predictions at the same points. This data set contains 653 WiMax measurements from the five BSs. Using the REMCOM Wireless Insight software, a scenario that models the transmitter antennas as generic sectors with the correct beamwidth, transmission power, location, and orientation is defined. The buildings are modeled using the STL data extracted from the Google 3D warehouse and placed on a flat terrain (the CU campus has little actual elevation change). REMCOM's proprietary "Full 3D" prediction method is used to predict a path loss value. Antenna models used are generic 120-degree sector antenna patterns without downtilt, and rotated to the correct position in the azimuth. Results are recorded in a proprietary, but parseable output format by the software.

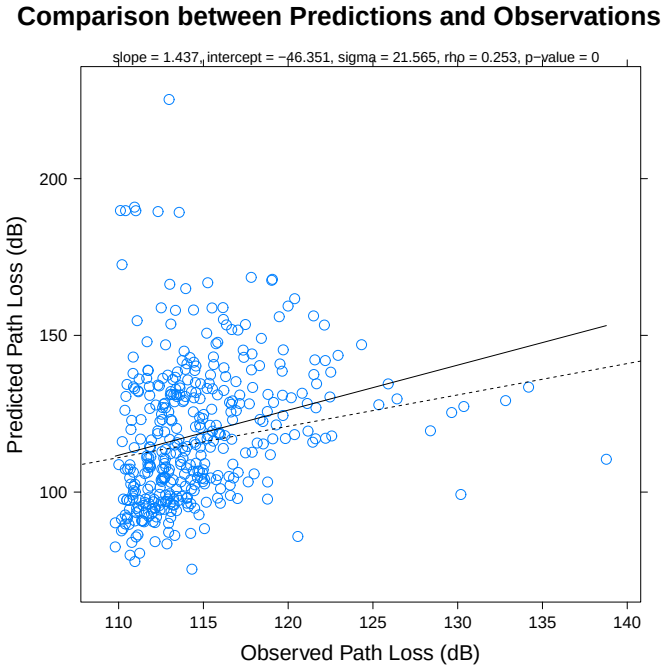
In sum, the predictions have little bearing to the observed values. The overall RMSE is 58.27 dB. The absolute error appears to be normally distributed, with a mean of 38.34 and standard deviation of 43.91. Figure 3.29 shows a point by point comparison of the predictions to the observations, and clearly only a weak correlation is present ($\rho = 0.253$ with $p - value \approx 0$), ruling out the hypothesis that the error could be from a systematic shift. If predictions where the REMCOM software refused to make a prediction and returned the noise floor value (presumably due to an error with knife-edge diffraction computation) are removed, the RMSE is reduced to 24.09 and the mean absolute error to 18.56. Despite being a well-regarded tool for wireless prediction and planning in general, the REMCOM software performs poorly in this scenario. Although, we cannot claim that this is necessarily a representative application (and, indeed the complexity of the building data may have negatively affected results), this does demonstrate the sort of errors that might be observed in a typical application of ray-tracing software using building models derived from crowd-sourced data. In this case, the ultimate performance is on the same order or worse than much simpler path loss prediction methods described above.

3.7.2 Case Study: EDX SignalPro and Rectilinear Building Models

In this scenario, the EDX SignalPro software was used with simple rectilinear building models. These building models were provided by EDX engineers, who have used the CU campus for testing their software, and were described by the engineers as typical of the building models many of their clients would use [238]. Unfortunately, the EDX software was unable to make use of the crowd-sourced building models, so a direct comparison between the results from the two software packages is not possible. Figure 3.30 plots the measurements versus the predictions. Compared to the previous scenario, the predictions here are actually better correlated with the observations. A Pearson's correlation coefficient of $\rho = 0.271$ with a $p - value \approx 0$ is observed. Similarly, a linear fit has a slope of 0.838 and an intercept of 6.019 indicating that the predictions fall roughly along the same line of the observations (with a 6 dB systematic shift). However, there is still substantial residual error, with a standard error (RMSE) of 11.372. As compared to the results with the REMCOM software, this is actually quite good, and on the order of the best untuned basic propagation models. With some tuning and correction from a few measurements, it is easy to imagine



(a) Each AP



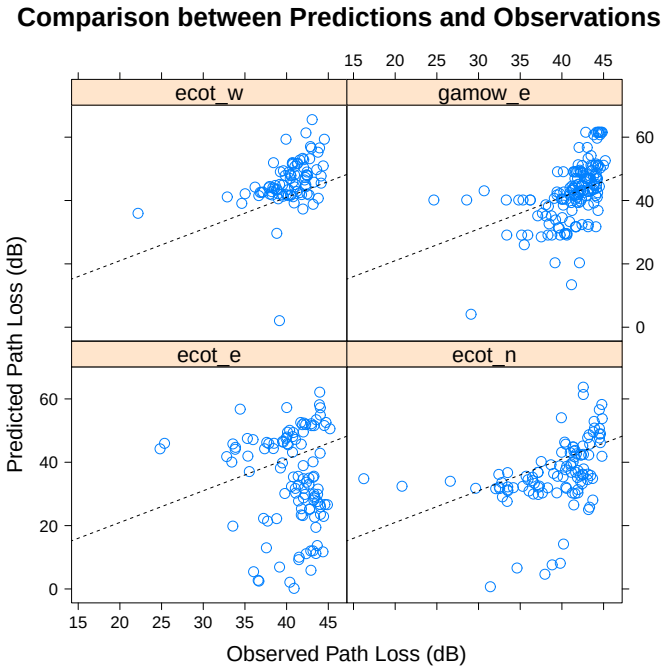
(b) All APs Combined

Figure 3.29: Correlation between predicted values and observed values using REMCOM ray-tracing software and WiMax data. The dotted line has a slope equal to 1, which the data points would fall upon if the predictions were perfect. Deviations from this line indicate the magnitude of error. Fit and correlation statistics are given for the aggregate (all APs) predictions. To simplify the plot, points where the prediction software refused to make a prediction have been censored, as well as locations where there was no signal observed.

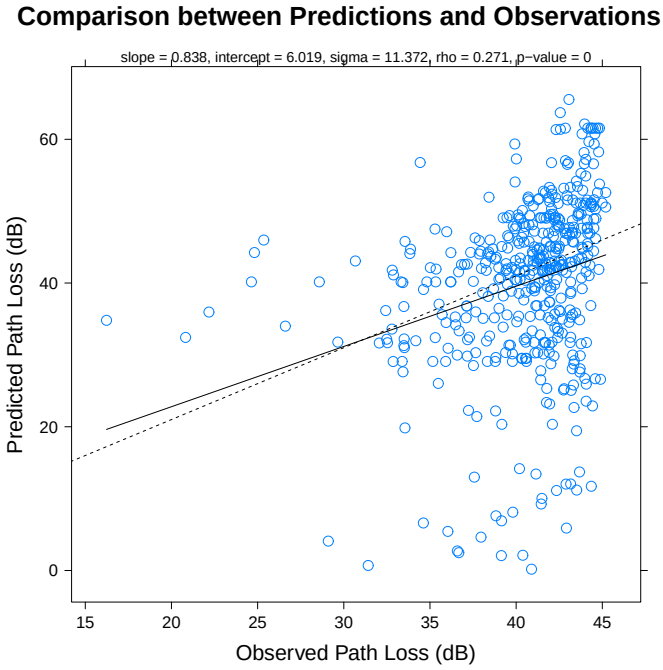
that this software could produce results of the same order of accuracy as those with an explicit data fitting or model tuning approach described in previous sections. However, this accuracy comes at the cost of increased data requirements. Determining the shape and height of every building in a study area to create a rectilinear building model may be very time consuming, even as compared to the time required to make some number of measurements.

3.7.3 Summary of Results

In the two scenarios studied, an unintuitive result was produced: the scenario with lesser building data fidelity outperformed that with higher resolution building models. This may be due to the fact that the complex polygons produced by the crowd-sourced building data produced many diffraction errors that were not present in the simpler data. Unfortunately, state-of-the-art ray-tracing tools are simply incapable of using high resolution building data collected from a LiDaR scan, and hence need substantial improvement in their efficiency and preprocessing algorithms to work with data of this fidelity. In an application where it is more costly to make direct measurements of the radio propagation than it is to gather data about the obstacles and buildings in the environment, then the use of these complex ray-tracing models may be justified. For basic planning purposes, their fidelity is likely sufficient. However, they do not appear to be substantially more accurate than a well-chosen simple path loss model (e.g., something from the Hata family). Some experts would argue that their fidelity is pendulously tied to decisions about how to model the diffraction and absorption of building construction materials (a classic example being buildings with radio-opaque glass, which may act as Farraday cages), and the choice of which subset of rays are used for calculation [142]. Although a great deal more work is needed to generally understand the relationship between the performance of ray-tracing approaches to path loss modeling and the fidelity of input data, these two case studies makes a compelling argument for the value of an increased focus on empirical coverage mapping as opposed to greater complexity in computation and environment modeling, whose performance in the general case is not well understood.



(a) Each AP



(b) All APs Combined

Figure 3.30: Correlation between predicted values and observed values using EDX ray-tracing software and WiMax data. The dotted line has slope equal to 1, which the data points would fall upon if the predictions were perfect. Deviations from this line indicate the magnitude of error. Fit and correlation statistics are given for the aggregate (all APs) predictions. To simplify the plot, points where the prediction software refused to make a prediction have been censored, as well as locations where there was no signal observed.

3.8 Discussion

This chapter has presented the first rigorous evaluation of a large number of path loss models from the literature using a sufficiently representative data set from real (production) networks. Besides providing guidance in the choice of an appropriate model when one is needed, this work was largely motivated by a need to create baseline performance values. Without an existing well-established error bound for these approaches, it is impossible to evaluate the success (or failure) of more complex approaches to path loss modeling (and coverage mapping). For the models implemented here, and the data sets analyzed, it is possible to say that *a priori* path loss modeling will achieve, at least, 8-9 dB RMSE in urban environments and $\approx$ 15 dB RMSE in rural environments. This is true almost regardless of the model selected, how complex it is, or how well it is tuned. And, this bound seems to agree with prior work at other frequencies in similar environments that have also produced results with RMSE in the neighborhood of 9 dB (e.g., [73, 60]).

Direct approaches to data fitting, such as a straight line fit to the log/log relationship between path loss and distance, produce a similar level of error: 8-9 dB for urban environments and $\approx$ 15 dB for rural environments. Fits of this quality can be obtained after only 20-40 measurements. Hence, whether a network operator does a small random sampling and basic fit, or carefully tunes an *a priori* model to their environment, they can still expect predictions that are only accurate to within 3 to 5 orders of magnitude. This result motivates continued work on more advanced methods and creates a well-defined measure of success for these more advanced models in terms of overall prediction accuracy: if a model can produce a coverage map where the variation (error) between the measurements and model is less then 12 dB, than we can say with confidence that it is outperforming an equivalent map generated using state-of-the-art *a priori* modeling routines. Moreover, a map with less than 8-9 dB error can be said to be better than can be expected with either hand tuned per-environment modeling or exhaustive measurement and explicit (straight line) fitting.

Among the most important outcomes of this work is a set of guidelines for researchers, which can help provide direction in the complicated landscape of path loss prediction models. As a general rule, when it is feasible to make direct measurements of a network, one should do so. It has been shown that a small

number of measurements can have substantial power in terms of tuning the models studied and in fitting parameters for basic empirical models. When it is not possible to make measurements of a network, the careful researcher should choose from standard well-accepted models such as Okumura-Hata or Davidson, which generally have the least systematic skew in predictions, and are among the best-performing models overall. In simulation studies, a repeated-measures approach is advocated, where stochastic models are used in a repeated-measures/Monte Carlo experimental design, so that a realistic channel variance can be modeled. For this application, the recent proposal of Herring appears to be a good choice, or for the greatest comparability, the Hata model with stochastic lognormal fading. Although there are a large number of models from which to choose, this work shows that in many cases the most important factors that a researcher should consider are having a realistic expectation of error, and choosing a model that enables repeatability and comparability of results.

Chapter 4

Large Area Coverage Testing

.

Over the past several years more and more cities, townships, and institutions have been deploying large scale wireless networks. On the largest scale, combination infrastructure and mesh networks are being used in municipalities to cover very large areas [122, 217, 95]. Many such deployments have been fraught with controversy around deployment motivations, performance expectations, and business models [228]. One possible explanation for these issues, offered by this thesis, is a failure to understand and effectively communicate performance expectations of the networks. With a robust and rigorous coverage testing methodology, many of these controversies and unfulfilled expectations are mapped into a clear and quantifiable problem and solution space. Indeed, the best way for a municipality to ensure that expectations are met is to be clear about the coverage and performance criterion of the network, and to ensure that this is tested in a thorough way.

As a first step towards developing a statistically robust method for coverage mapping, this section approaches the sub-problem of "coverage testing", that is, making a strong statement about the *percentage* of area within a region that is covered by a given network. Coverage testing has its own important applications in contractual verification. Because rigorously solving the coverage mapping problem involves solving the coverage testing problem first, this chapter begins here. The method developed in this chapter relies on basic and well-established statistical methods, including a selection of a Simple Random Sampling (SRS) of points, and testing via the Binomial Theorem. Combining these robust statistical methods with a data

[0] Work in this chapter has appeared in [171, 168]. Data collected for the experiments in this chapter has been made publicly available at [170].

collection methodology that is appropriate for sampling the radio environment is able to produce a simple method for coverage testing that requires a *minimum amount of work* required to make an accurate statement about coverage.

Most coverage and performance testing of large networks is carried out by contractors (e.g., [239, 4]) who use proprietary and sometimes nonrigorous techniques to perform their tests. By comparison, the methodology presented here not only comports with Occam's razor, it is also based on low cost and readily obtainable commodity hardware. Additionally, all techniques are passive, requiring no more access to the network than any casual observer would have. Because the methodology is simple and the hardware inexpensive, it may even be possible for some testing to be carried out by institutions and municipalities themselves. At the very least, simple and well-defined approaches to coverage testing will serve to encourage transparency in the testing of contractors, which will go a long way to making results easier to interpret and validate.

4.1 Method

The complexities of the wireless medium require that measurement strategies are approached carefully. We want to make experimental assumptions that are enlightened with respect to both the properties of RF propagation [183] and of infrastructure wireless networks [120]. This section outlines domain appropriate guidelines for coverage testing. The following section will apply these guidelines to develop a practical coverage testing methodology for a municipal wireless network in Portland, Oregon.

Signal Strength Alone Is Not Enough

Bidirectional communication in wireless networks requires a symmetric concept of a link: *just because a client device can hear an AP does not guarantee that the AP can hear the client device* [120]. In practice, wireless APs are often much more powerful than wireless clients. A typical outdoor AP may include a 400 mW radio connected to a high gain antenna, resulting in an equivalent isotropically radiated power (EIRP) as high as 4 W[1] . In comparison, a common client device might have a 30mW radio attached

[1] The Skypilot-brand radios used in Portland, Oregon, for instance, have a transmit power of 400 mW and a 7.4 dBi omnidirectional antenna, resulting in an EIRP of 2.2 W (33.4 dBm)

to a meager antenna (2-5 dBi is common in our experience) providing an EIRP of closer to 17.8 dBm (60.26 mW). Although the AP's antenna will provide gain on receive as well as transmit, this cannot make up for the clear asymmetry in power and sensitivity of the two devices, which results in many situations where a client device can see a strong signal from an AP, but is unable to get its communications back to the AP[2] . Therefore, Neither RSS, nor SNR are appropriate measures of link quality [20] alone. By themselves, they form a poor basis for inferring about usable coverage. If one wants to use distance, SNR, or any other variable alone as a single value indication of link quality, a relationship should be experimentally derived based on the appropriate environment and the equipment. If this is done with acceptable thoroughness, it may produce coverage extrapolations that are acceptable using this value alone.

Environmental Diversity

As discussed in [183], the quality of a wireless signal can vary substantially due to the location and the characteristics of the environment in which it is measured. Due to this, any scheme that purports to quantify the performance or coverage of a wireless network must give careful consideration to where measurements are made so that they do not skew the results in one direction or another. It might not be safe to use information drawn from one wireless environment to make conclusions about another—any such extrapolations should be treated with extreme skepticism.

Variation in Hardware

Wireless networking hardware varies greatly. Principally, variations in receiver sensitivity, transmit power, and antenna gain are most troublesome. Any equipment used in testing should be convincingly representative and should be carefully calibrated. If nonrepresentative hardware is used, then a normalization procedure should be adopted and independently confirmed. In all likelihood the easiest approach here is to use representative hardware and avoid the onerous task of normalization.

Other Operators, Other Networks

Measurements of a live network must consider effects of other users on that network, and of interference from neighboring, but unrelated, networks. The former can be addressed by testing the network during

[2] This is especially a concern in the case when a user is indoors and the AP is outdoors; in such cases it may simply be impossible to achieve high quality of service without using a more powerful antenna on the client side.

a time when it is not in use. The latter is a concern that a network designer must address when they deploy their network (choosing channels to minimize cochannel interference, etc.).

Application Layer Testing

The best way to model the usability of the network is to approach problems with the perspective of real use cases. This means that when we do a point test of network quality we gain the most by doing application layer tests, such as throughput and latency testing in addition to low level tests (such as signal strength and noise level). Ideally, the endpoint for such tests would be very near the endpoint of the network to remove effects from outside the network.

Sampling Design

For a small network, it may be feasible to measure the entire expected coverage area. However, this quickly becomes intractable for larger networks. Choosing an appropriate statistical sampling design is crucial to draw a useful conclusion from the results. Although there are many approaches to spatial (sometimes called regional) statistical sampling, not all are appropriate for the problem. Section 5.2.1 provides a discussion of classic sampling schemes and the tradeoffs involved in design selection. Because it is least likely to be aligned with sources of error and is easy to implement and put to use, SRS is the sampling strategy advocated here.

Dealing with Unreachable Points

It is inevitable that when testing sample points in any well-designed spatial sampling scheme, some points will not be reachable. They might, for instance, be in the middle of a freeway, or a river, or on private property. These points should be measured on a best effort basis as close to the original sample point as possible and the deviation should be carefully documented. Often, an assumption of spatial sampling is that values at geographically close points are similar. While the wireless medium is highly variable, with the exception of extreme shadowing scenarios, it is unlikely that two close points will differ substantially in coverage. Hence, making a best effort measurement in some small set of pathological cases is unlikely to significantly bias results. In the case that it does, careful documentation will be rewarded.

Sample Size

The required sample size for a certain confidence interval is dependent on the variability of the results.

If an SRS is used, points can be tested up until the confidence interval narrows to the desired value.

Temporal Variability

Because the behavior of wireless networks has been shown to be variant in time [20], long term tests are required to determine temporal variability in network performance. Ideally, these tests would run for as long as possible and the testing points would be distributed using the same random sampling technique used for coverage testing. However, long term testing introduces some logistical complexities that may require some compromises. For instance, the test device is likely to need AC power and a good vantage from which to test. It is unlikely an SRS will choose positions that are appropriate for long-term tests. As such, it may be necessary to deploy long term test devices in locations where the testers can acquire permission and access at the sacrifice of proper sampling design. In any case, given choice of locations, the more convincingly representative the subset, the more useful the results will be.

4.2 Case Study: Portland, Oregon

In September of 2005, the city of Portland, Oregon issued a Request for Proposals (RFP) to build and operate a “citywide broadband wireless system”. In April of 2006, the city chose MetroFi (Mountain View, California) as the winning bidder, and in the following summer the city and MetroFi signed a nonexclusive license agreement. Thereafter, MetroFi began to deploy their network in preparation for a December 2006 launch of a Proof of Concept (POC) network, as called for in the agreement. The deal was structured such that the POC network would first be built and afterward an independent third party would test it. When the city was satisfied that the POC network met its performance criteria, it would issue a certificate of acceptance. Specifically, the contract stated that the network should be able to provide a connection to at least 90% of the outdoor POC area (defined as all locations within 500 feet of the 72 APs). Further, for a given stationary connection, the network should support a 1 Mbps downstream/256 Kbps upstream throughput, have 99% availability, and a total within-network latency of 100 milliseconds. During this POC testing phase, an independent analysis of the network coverage was conducted.

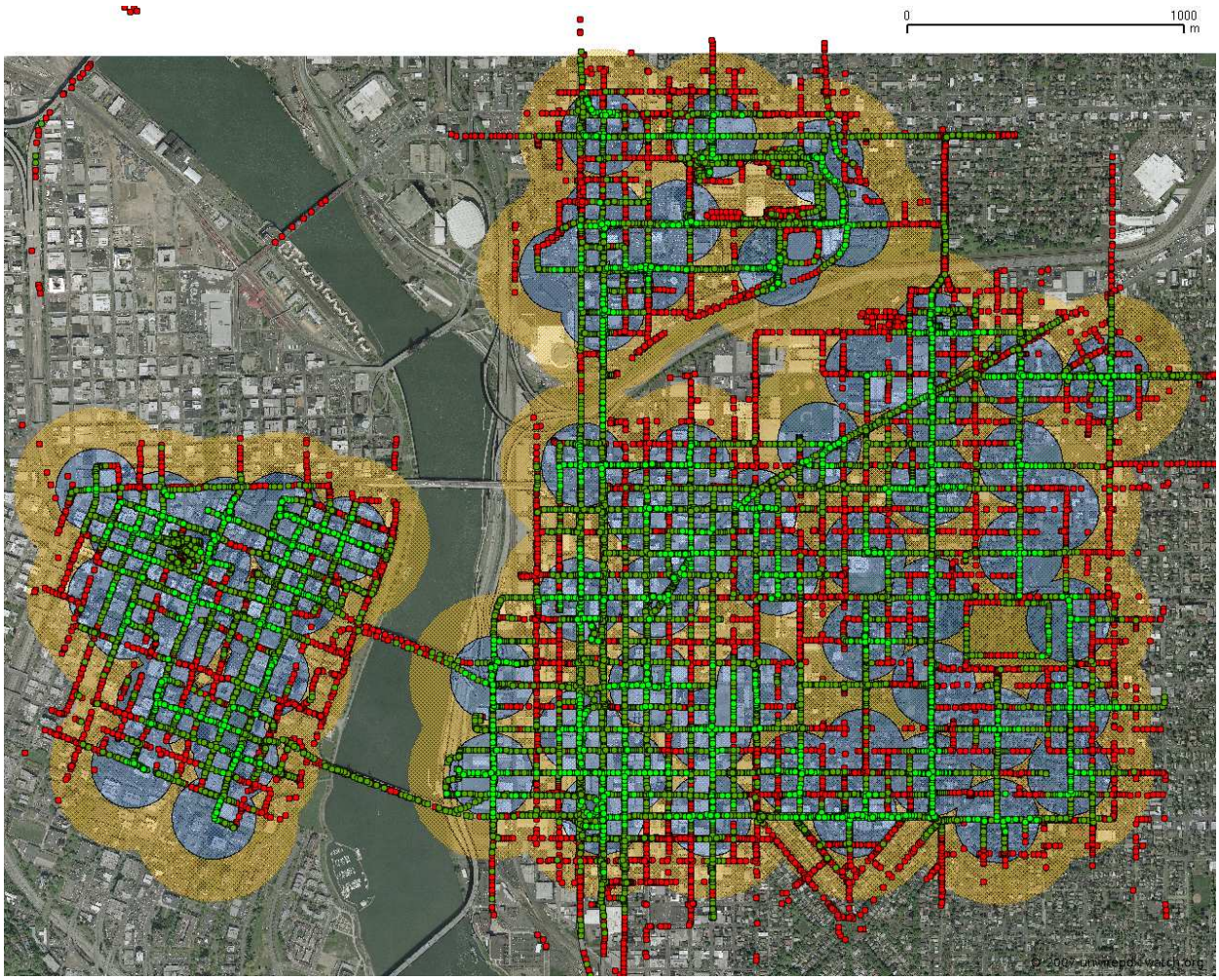


Figure 4.1: Signal strength from APs in the POC area. Lighter dots (green) indicate stronger signal.

4.2.1 Method

Because the tests were carried out without any access to the network infrastructure, the first task was to locate the APs in the POC area and obtain signal strength measurements over the entire area. To this end, every publicly accessible street was driven, collecting signal strength measurements using a battery powered embedded computer with an external 7 dBi omnidirectional antenna and a GPS device. Figure 4.1 plots the measured signal strengths. This data was used to triangulate the position of the APs. Not surprisingly, as other researchers have shown that signal strength is poorly correlated with distance [20], a satisfactory level of precision was unable to be obtained. To obtain the desired precision, triangulation was used to locate each AP, and then a reading was taken with a handheld GPS device directly under the AP. To compress this data set slightly, precision of GPS coordinates was truncated to five significant digits, which has the effect of grouping data points within a 0.74 m circle.

4.2.1.1 Sampling Metholodogy

From the list of 72 MetroFi APs that were considered to be in the POC network, a bounding box in latitude and longitude[3] was constructed extending 1,000 feet beyond the extremities of the AP locations. Because it was expected that many locations in the bounding box would fall outside of the POC areas, and because it was not certain how many locations we would be able to measure, an excessive sample of 1,001 locations was computed using a random number generator such that each location in the bounding box had an equal probability of being chosen. Locations not within 1,000 feet of an access point were immediately excluded. Each remaining location was plotted against orthoimagery using Google Maps. If the location fell in the Willamette River, was inside a building, or was not practically reachable, it was also excluded. Ultimately, the first 250 locations in the sample of 1001 were either excluded on the basis of the criteria above or were visited and measured (see Figure 4.2). It was decided to stop after surveying 250 points because the results had sufficient statistical power at that point.

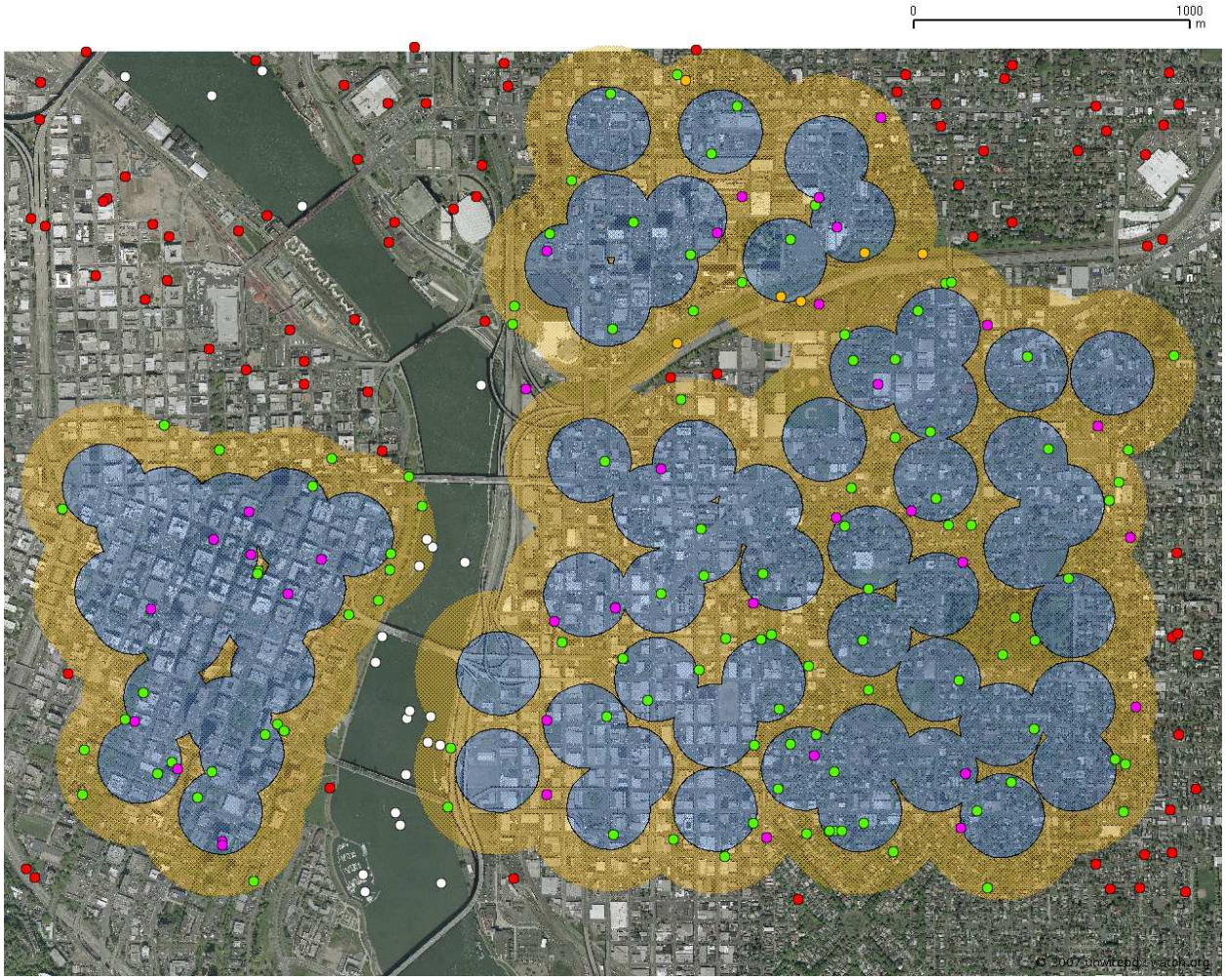


Figure 4.2: Random locations and their categorization. Green (light grey) dots were tested, purple and orange (grey) were points within the POC that were excluded because they were inaccessible, and red (dark grey) were excluded because they were not within the POC.

Figure 4.3: Testing apparatus. A battery powered Netgear WGT634u wireless router outfitted with a GPS device, USB storage, speakers, and an enable key.

4.2.1.2 Measurement Apparatus and Procedure

To act as a coverage point tester, a low cost single board computer (a Netgear WGT634u router) was combined with a reliable Linux-based firmware (OpenWRT GNU/Linux), a lithium-ion battery, USB GPS receiver, and USB compact-flash storage. In addition to the mandatory components, a USB sound card and a pair of small speakers were used to "speak" status updates along with a small Bluetooth USB dongle that was used as an "enable key"[4] . All together, this testing apparatus cost less than $200 USD to build. Additionally, the Atheros 5213 802.11b/g radio and attached 2 dBi omnidirectional antenna fulfilled the requirement that the testing apparatus be representative of a typical client device. The test device was rigged to be freestanding at six feet off the ground so that the operators would not interfere with the measurements. When enabled, the test device was programmed to carry out a series of tests. The outline of the testing

[3] All latitude/longitude coordinates are with respect to the WGS84 ellipsoid, unless otherwise noted.

[4] A small test was conducted using a WiSpy spectrum analyzer to test whether the Bluetooth device was radiating (and thus causing interference with the test device) when used this way. It was concluded that the bluetooth dongle does not emit noticeable radiation when it is not in use.

procedure is given in algorithm 1.

Algorithm 1 Point testing procedure

1: Disassociate
2: Try to associate with an AP for 60 seconds
3: Record information about the physical layer (BSSID, Signal, etc.)
4: Try to obtain a DHCP lease by sending up to 10 DHCP requests
5: Attempt to pass traffic to the Internet, if unable, bypass the captive-portal
6: Test latency and loss using ICMP ping
7: Test downstream throughput with a 1MB file, and a 5MB file
8: Test upstream throughput using ttcp
9: Store the contents of the ARP table
10: Store some statistics about our test device (memory and CPU utilization, etc.)
11: Perform a traceroute to an internet host to record routing topology

Standard Unix tools were used: ttcp, to test upstream throughput; Internet Control Message Protocol (ICMP) ping, to test latency and loss; and wget, to test downstream throughput. A small script was used to bypass advertisement traps[5]. It was also found to be necessary to use several watchdog scripts to check for a lost association, GPS issues, and stalled tests (for example, ttcp has a tendency take a very long time on unstable connections). Depending on the results, a random location test might take anywhere from about 60 seconds (the length of time that was waited for an association) to around 7 minutes. In addition to these steps, GPS position and time-stamp were also recorded throughout the test.

The results of each test were stored on the USB storage device. At the conclusion of the tests the results where retrieved and analyzed. In the analysis each visited location was categorized according to the states in table 4.1. By categorizing points by their success state in table 4.1, the set of test points can be treated as a binomially distributed Bernoulli trial—states 1 to 5 indicating failure and state 6 indicating success. Hence, classic binomial hypothesis testing can be used to analyze the results.

In addition to coverage testing, three more test devices were deployed for long term tests. As noted earlier, finding appropriate locations for long term testing poses some logistical challenges. Thus, the devices were positioned at the best locations permitted for use, and not at positions dictated by a simple random sample. At each location, continuous tests were run for a minimum of a week, collecting throughput, la-

[5] The “free” public MetroFi network was configured to inject banner advertisements into Hypertext Transport Protocol (HTTP) results and periodically redirect HTTP requests to a full-page advertisement. These advertisement traps, if not otherwise bypassed, would have interfered with the downstream throughput test, which involved downloading a fixed-size file over the HTTP protocol on port 80.

State	Description
1	Could not associate
2	Lost association mid-test
3	Could not get a DHCP lease
4	Could not pass traffic
5	Performance below specified
6	Success

Table 4.1: Point test state categorization

tency, and link-quality information. The hardware and test methods here are identical to those used for coverage testing.

4.2.2 Results

The first task in analyzing the results from the coverage tests is to infer a coverage percentage and a confidence interval for this inference. Figure 4.4 shows the p-value for an exact binomial test as the radius of points from the nearest AP changes and the hypothesized coverage percentage changes. Notice that any area where the p-value is less than $\alpha = 0.05$ is rejected, which is essentially all of the combinations outside the prominent "ridgeline". In effect, the width of the ridgeline at any radius provides the 95% confidence bounds for the coverage percentage. For instance, at 150 meters, there are acceptable p-values only between about 50% and 70%. The contract required 90% coverage within 500 feet (approximately 150 meters) of each AP. The measured percentage covered was 44.4% overall and 63.46% within the 500-foot radius. The probability of the coverage requirement being satisfied given the overwhelming evidence against it is one in 4,451,872. According to this map, the only radii that can achieve a coverage criterion of 90% are 50 meters or less (where the p-value is near 1). It is worth noting that some of the results here differ substantially from those of the contracted company, [4]. A discussion of these differences is outside the scope of this thesis, but can be found at [169].

It should be noted that this value, 44.4%, indicates that less than half of *locations* within the coverage area are expected to be able to achieve a connection at the performance required by the contract. Additionally, if poorly performing locations i.e., locations providing a connection with slower throughput or higher

latency than required by the contract, are included, it can be said that at a 95% confidence level, the percentage of locations acheiving *any connection* is between 36.08% and 54.77%. From the perspective of municipalities hoping to deploy a wireless network for the purpose of automated meter reading and other such applications, these numbers are fairly dismal and further serve to highlight the fact that it is essential that requirements are well specified and tested to ensure that both the needs of the network operator, and that of the institution or city are met.

Although the network in Portland does not meet the coverage criterion defined in the contract, it is not clear that this coverage criterion was formulated in the best possible way. Instead of defining an arbitrary POC area as a certain radius from each AP, a more useful metric would be to define a (more conservative) percentage goal for the entire region to be covered. Additionally, the contract should be straightforward about the way this coverage will be tested in terms of sampling and performance goals. In the case of the network in Portland, at 44.4% it is still very low, indicating that the network operator should seriously consider increasing AP density[6]. Moreover, since this testing was conducted exclusively outdoors, it can be at best looked at as an extremely optimistic estimate of indoor coverage.

Interestingly, signal strength is normally distributed among points where it was possible to associate with the AP. A Shapiro-Wilkes test gives a p-value of 0.297, i.e., unwilling to reject the null hypothesis that the samples are normal. Overall, signal is highly variable among those points that had successful connections, providing a mean value of -63.06 dBm and standard deviation of 9.63 dBm. Among those points where association was successful, but the test failed somewhere upstream, the mean signal strength is -77.13 dBm with a standard deviation of 5.80.

State and signal are reasonably linearly correlated, showing a correlation coefficient of 0.47. This correlation is very strong if we assume signal strength -95 dBm (essentially, the noise floor) for those trials that failed to associate (the coefficient is 0.90 in this case). Distance, however, is not well linearly correlated well with state or signal (correlation coefficient is -0.36). Information about the performance of the network was collected at points that were successful. Averaging across the random sample provides an “expected

[6] The hardware vendor (SkyPilot [225]) claimed that this particular network was underdeployed relative to their recommendation[3].

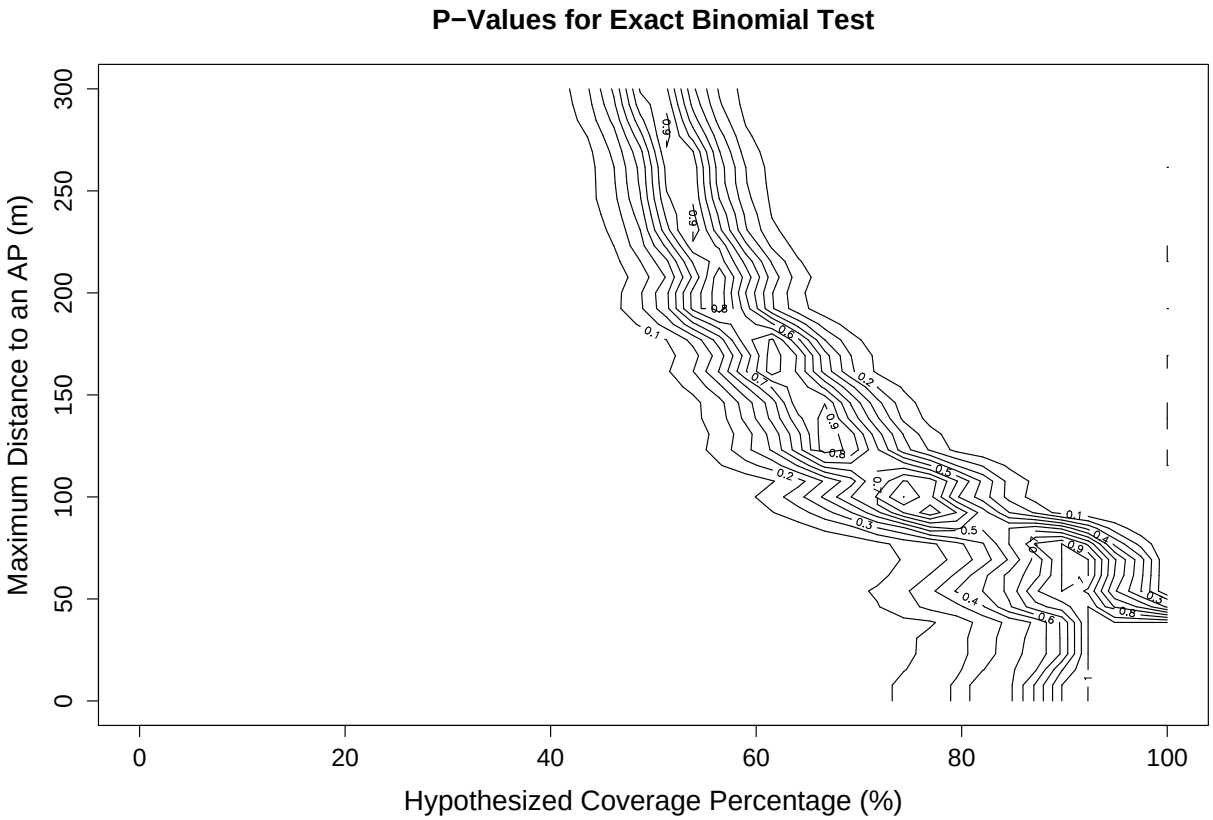


Figure 4.4: Contour map of p-values for an exact binomial test as a function of maximal distance to an AP (i.e., only concerning samples within some radius) and hypothesized coverage percent. p-values below $\alpha = 0.05$ reject the null hypothesis that the hypothesized coverage percent is possible given the observations.

Area (ft)	N	Down/Up Throughput (Kbps)		Latency (ms)		Loss (%)		Signal (dBm)	
		Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
<250	7	1672.4/373.28	1281.3/83.235	95.700	72.319	5.7143	15.119	-50.857	3.7607
<500	16	1508.7/373.42	1002.8/79.181	105.15	69.808	3.125	10.145	-57.938	8.4417
>1000	27	1437.2/370.52	875.72/74.682	97.459	59.344	3.33	8.77	-59.333	8.6425
>500	11	1333.1/366.30	657.23/71.159	86.273	40.182	3.6364	6.7420	-61.364	8.9249

Table 4.2: Random sample performance summary

view" of performance for those locations with a usable connection. These statistics are summarized in table 4.2.

Although the long term tests are not clearly representative because of logistical limitations, a large amount of continuous data at three locations within the POC area was still able to be collected. Site A was collected on the first floor of a house in a residential area, very close to an AP; site B was collected on the second floor of an office building on the edge of downtown Portland; site C was collected in the window of a fifth-floor office in the heart of downtown Portland. A summary of some of the statistics from these tests is in table 4.3. In terms of the performance requirements of the Portland network, all three locations passed.

One interesting observation, however, is that the performance can vary highly as a function of time of day. Figure 4.5 plots packet loss for site C as a function of time of day. Notice that site C, which was in a densely populated area (both in terms of people and wireless networks), exhibits large packet loss during the bulk of the typical business day. It is hypothesized that this is a result of internetwork interference. If nothing else, this plot should be yet another warning to network operators that interference from neighboring and third-party 802.11x wireless networks must not be neglected in the design and performance expectations of future networks.

Site	Duration (h)	Disassoc. Probability	Mean Percent Packet Loss	Percent Packet Loss Std. Dev
A	456.44	0.00149	1.562%	4.789%
B	173.83	0.00106	2.549%	7.418%
C	197.53	0.00449	33.031%	28.983%

Table 4.3: Summary of a selection of long-term test statistics.

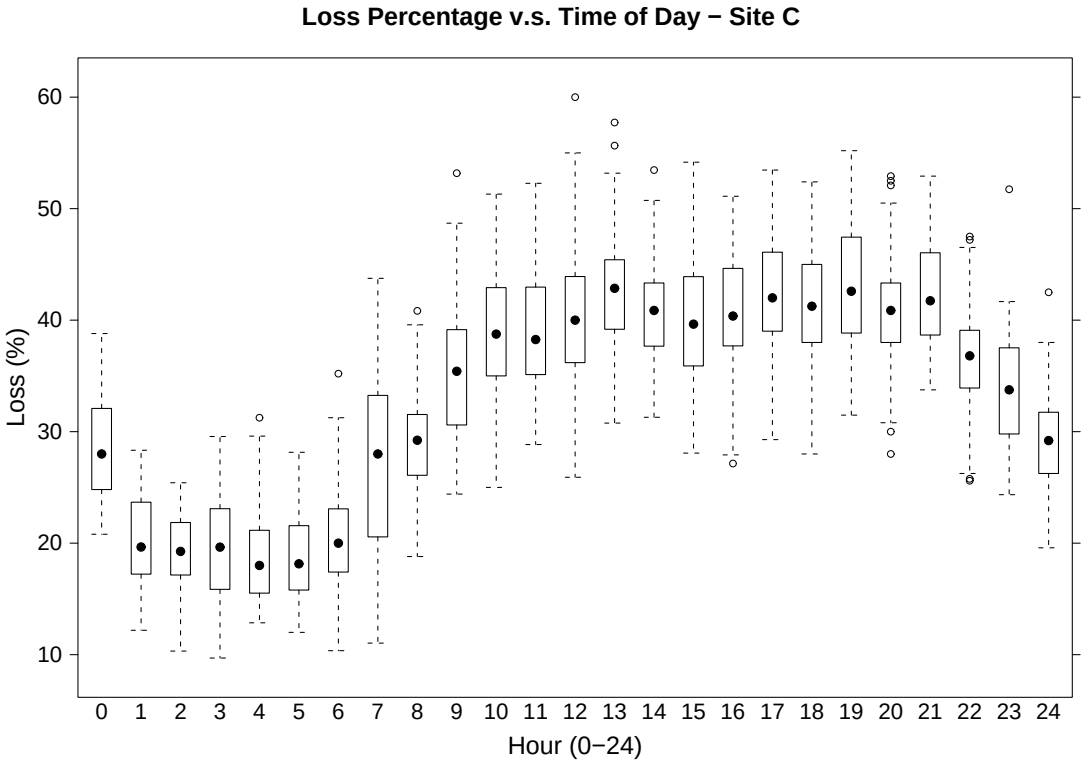


Figure 4.5: Packet loss for long term test at site C as a function of time of day. Measurements are averaged across days and bucketed per hour.

4.3 Discussion

This chapter has outlined a simple but powerful method for coverage and performance testing of large-scale wireless networks. The proposed method utilizes a random sample of points within the coverage area to make inferences about *usable coverage* and expected performance. For test results to be meaningful, it is crucial that an appropriate spatial sampling design be paired with a testing approach that both considers the perspective of the user and the complexities of the wireless medium.

This testing method was applied to a large municipal wireless mesh network in Portland, Oregon and results from that study have been presented. As similar networks continue to proliferate, having a practical and effective method to test them is vital to their success and to achieving a rational way of communicating expectations. The lessons learned in this study can be applied directly to developing appropriate methods for the larger problem of coverage mapping, the topic of chapter 5.

Chapter 5

Geostatistical Coverage Mapping

This chapter outlines an approach for robust coverage mapping using principled spatial sampling and geostatistical interpolation ("Kriging"). The chapter begins by providing some background on geostatistics for the uninitiated, since understanding the problem definition and assumptions are necessary to motivate the approach. Then, classic spatial sampling schemes are described along with more advanced multi-phase optimized sampling schemes (similar to those presented in chapter 8). Section 5.4 will walk though the fitting and mapping method proposed here as well as provide explanations and examples for the approach taken. This will set up the next chapter, 6, which descibes the application of these methods in two case studies mapping the coverage of production networks.

5.1 Geostatistics in a Nutshell

There are a number of textbooks that cover the topic of geostatistics in depth. The encyclopedic treatment by Cressie is a fine starting point [55]. However, Wackernagel's text [233] is more approachable for most topics and [118] provides a concise discussion of state-of-the-art and advanced geostatistical modeling techniques. Other texts that are less lucid but still worth mentioning are Ripley's very dense books [197, 196] and the thorough treatments in [46] and [104]. Also worth noting is [124], where Krige and Kleingold provide a history of the development of the field throughout the 1950's, 1960's, and 1970's.

5.1.1 A Random Field Called 'Z'

If we assume that there is a random field being modeled called Z, then the value of that field at a point in space $\mathbf{x}$ is $Z(\mathbf{x})$. The field can be defined in any dimension, but it is typically assumed that $\mathbf{x} \in \mathbb{R}^n$ with $n = 2$ or $n = 3$. The value at any point can be defined as the field mean (μ) plus some error ($\epsilon(\mathbf{x})$):

$$Z(\mathbf{x}) = \mu + \epsilon(\mathbf{x}) \tag{5.1}$$

5.1.2 The Variogram

Central to geostatistics is the variogram, a function that models the variance between two points in space as a function of the distance between them ($\mathbf{h}$). In the case of grid-sampled fields, the distance between measurements is a fixed lag distance. Randomized and optimized sampling schemes produce variable lag distances. The theoretical variogram, γ, is typically written as a function of the expected value of the squared difference between a given point value and a point some lag h way:

$$\gamma(\mathbf{h}) = \frac{1}{2}E[(Z(\mathbf{x}+\mathbf{h}) - Z(\mathbf{x}))^2] \tag{5.2}$$

If it is known that the field is second order stationary (i.e., a measurement at the same point will not vary with time and the difference between two measurements at the same two points will not vary with time), then the covariance function (correllogram) is defined as:

$$C(\mathbf{h}) = E[(Z(\mathbf{x}) - \mu)(Z(\mathbf{x}+\mathbf{h}) - \mu)] = C(\mathbf{0}) - \gamma(\mathbf{h}) \tag{5.3}$$

However, second order stationarity is probably not a safe assumption for the radio environment (not without some effort to correct for temporal variation anyhow). With some set of measurements, an empirical variogram can be defined as the sum of squared differences for each observed lag distance h_i:

$$\gamma'(\mathbf{h}_i) = \frac{1}{2n}\sum_{j=1}^{n}(z(\mathbf{x}_j + \mathbf{h}_i) - z(\mathbf{x}_j))^2 \tag{5.4}$$

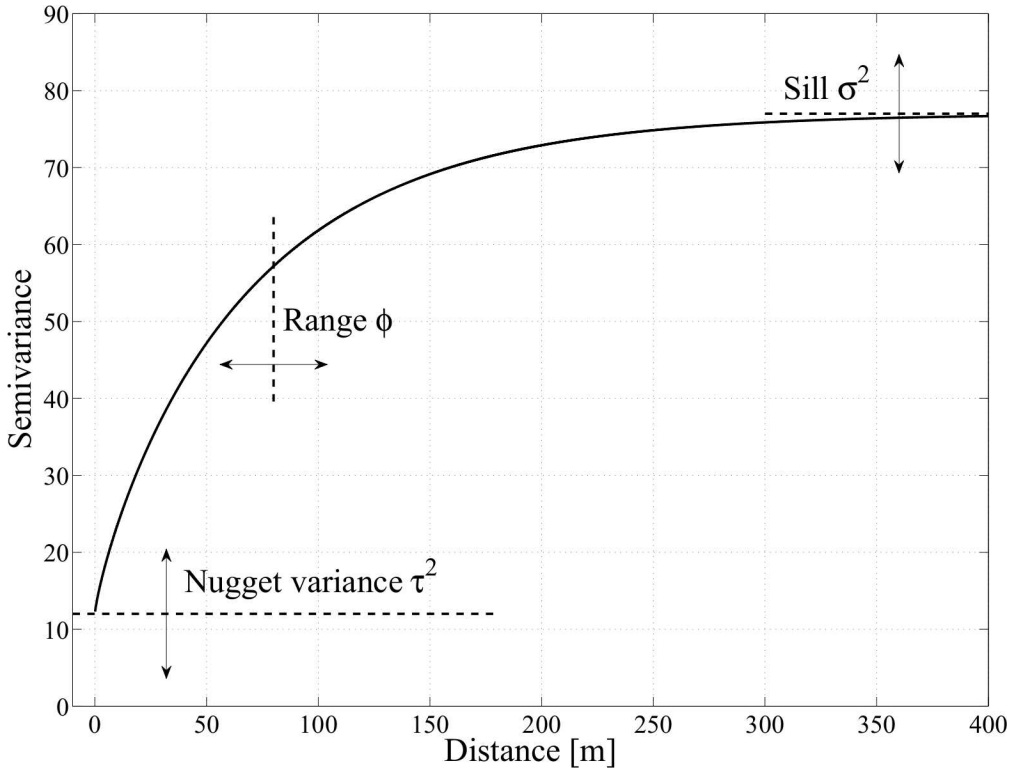


Figure 5.1: Explanation of variogram model parameters using the Matérn model as an example. Figure taken from [236].

A typical problem is to fit a variogram (or correllogram) given some number of measurements. There are a number of models that can be used for fitting. One example is the exponential model:

$$\gamma_{exp}(h) = \tau^2 + \sigma^2(1 - e^{-h/\phi}) \tag{5.5}$$

Figure 5.1 shows how the parameters of this model effect the shape of the fitted variogram. τ^2 is known as the nugget variance and is used to model discontinuity around the origin, by increasing or decreasing the starting threshold. It is so thusly named because this parameter helps model the likelihood of rare minerals ("nuggets") in geological surveys. σ^2 is known as the sill because it sets the maximum value of the semivariogram. Larger values of σ will increase the level at which the curve flattens out. Finally, the parameter ϕ acts as a scale and affects the overall shape of the curve. The value of ϕ determines the rate at which variance is expected to appear as a function of distance (lag) between points. There are a number of other models, such as the Gaussian, Cauchy, and Matérn models, which may or may not be the best fit depending on the data. [159] provides an accessible introduction to variogram fitting by walking through an example fitting. In that work, Olea discusses the pros and cons of various permissible functions, efforts to remove trend, nested models and anisotropic methods. In addition to the classic models, more advanced (and nested) models can be used so long as they are permissible. For instance, [178, 113] discuss the (optionally damped) cosine Hole-Effect model which is able to capture periodicities at varying scales within the semivariogram.

Variogram fitting can be carried out using a variety of methods. Historically, the method of moments has been used to derive a fitted variogram. However, Maximum Likelihood Estimator (MLE) and Least Squares (LS) methods have been used more recently with some substantial success. In [163], Pardo-Igúzquiza comes out as a proponent of the MLE method, claiming that its requisite assumption of an underlying Gaussian distribution is reasonable, and supports these conclusions with simulations. In [162], Pardo-Igúzquiza describes a software package to fit the variogram to one of several models using this MLE method, along with a modified ("restricted") MLE approach that avoids errors from simultaneously estimating the drift and covariance parameters of the variogram. In [112], Jian *et al.* argue for a Weighted Least

Squares (WLS) approach and suggest that the Akaike Information Criterion (AIC) be used as a goodness of fit metric. Later work by Lark [127] gives a more rigorous comparison of these two methods and finds that for both simulated and real data sets, the method of moments and MLE fitted models are not substantially different. Both methods are susceptable to distributional skew and outliers. However, for some specific cases, each approach outperforms the other. For instance, when nugget variance is relatively small and the correlation range of the data is large, method of moments performs better. In sum, Lark recommends that fits be made with both methods, and the resulting modes compared.

5.1.3 Kriging

"Ordinary" Kriging is an interpolation technique that predicts the unknown value at a new location ($Z(\mathbf{x}')$) from the weighted known values at neighboring locations ($\mathbf{x}_i$):

$$Z_K(\mathbf{x}') = \sum_{i=0}^{n} w_i Z(\mathbf{x}_i) \tag{5.6}$$

and, to determine the optimal weights ($\mathbf{w}$), we must minimize the estimation variance σ_E^2:

$$\sigma_E^2 = E[(Z_k(\mathbf{x}') - Z(\mathbf{x}')^2] \tag{5.7}$$

with

$$\sigma_E^2 = -\gamma(\mathbf{x}' - \mathbf{x}') - \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \gamma(\mathbf{x}_i - \mathbf{x}_j) + 2\sum_{i=1}^{n} w_i \gamma(\mathbf{x}_i - \mathbf{x}') \tag{5.8}$$

which leads to the following system of equations:

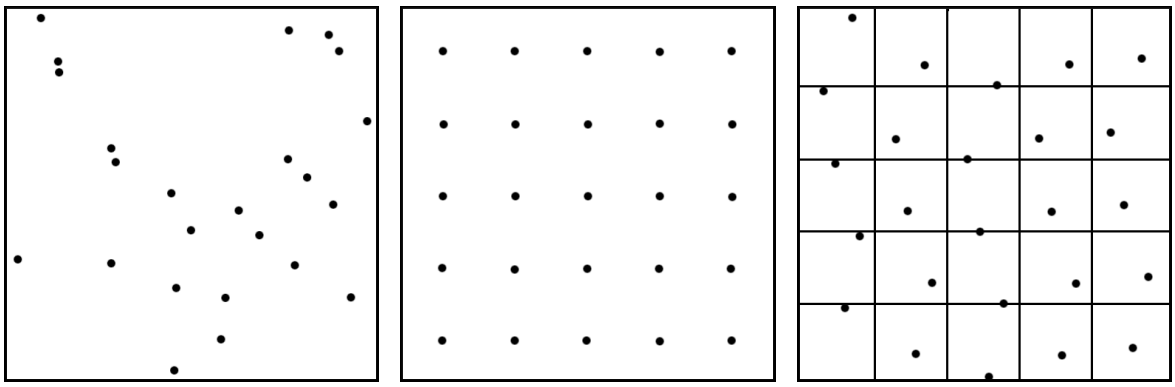
$$\begin{pmatrix} \gamma(\mathbf{x}_1 - \mathbf{x}_1) & \cdots & \gamma(\mathbf{x}_1 - \mathbf{x}_n) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \gamma(\mathbf{x}_n - \mathbf{x}_1) & \cdots & \gamma(\mathbf{x}_n - \mathbf{x}_n) & 1 \\ 1 & \dots & 1 & 0 \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_n \\ \mu \end{pmatrix} = \begin{pmatrix} \gamma(\mathbf{x}_1 - \mathbf{x}_0) \\ \vdots \\ \gamma(\mathbf{x}_n - \mathbf{x}_0) \\ 1 \end{pmatrix} \tag{5.9}$$

where μ is called the Lagrange parameter. This interpolation is "exact", meaning that $Z_K(\mathbf{x}') = Z(\mathbf{x})$ if $\mathbf{x} = \mathbf{x}'$. This approach can be used in mapping by Kriging the value at each pixel position.

The quality of an interpolated field depends on the goodness of the fitted variogram (γ). In addition to this, there are a number of different ways to adapt Kriging to a specific data set. Anisotropic corrections are of particular interest for coverage mapping. This approach assumes that the field may require different statistics (i.e., a different variogram and possibly fitting method) in different directions from some point. There is also an entire branch of statistics dealing with multivariate analysis (i.e., co-Kriging). Depending on the importance of the time dimension, for instance, these directions may be of particular interest.

5.2 Spatial Sampling

Despite the many measurement-based approaches to path loss prediction and coverage mapping, no single work has looked at the important questions of *where* these measurements should be made and how many of them are needed. These questions of *where* and *how many* are at the center of this thesis. This section provides background on spatial sampling.



(a) Random Sample (b) Systematic Sample (c) Systematic Unaligned Sample

Figure 5.2: Examples of sampling schemes

5.2.1 Classic Sampling

Choosing an appropriate sampling scheme is exceptionally application dependent. The shape and variance of the field, as well as domain-specific knowledge about the process being modeled, must all be considered when selecting a sampling strategy. In [214], Stehman proposes a useful taxonomy of tradeoffs

1	Satisfies probability sampling protocol
2	Is simple to implement and analyze
3	Has low variance for estimates of high priority
4	Permits adequate variance estimation
5	Is spatially well distributed
6	Is cost effective

Table 5.1: Requirements of an appropriate sampling design

for spatial sampling schemes that is reproduced for reference in table 5.1.

SRS is the classic approach used in many spatial sampling problems. An example is given in figure 5.2(a). It is simple, straightforward, and well understood. In particular, SRS is convenient in that any consecutive subset of a simple random sample is also a simple random sample itself. This means that one can create a sample of points, and then test them consecutively until a statistical significance criterion is met.

Competing sampling schemes might include systematic (see figure 5.2(b)), systematic unaligned (see figure 5.2(c)), or stratified. Systematic sampling can be dangerous because it risks alignment bias. It is admittedly tempting when measuring coverage to align samples along an obvious geographic feature of cities, i.e., streets. However, this thesis argues that aligning samples along streets risks highly biasing results. Aside from degenerating to a type of one-sided stratified sampling, streets also have the capacity to act as RF waveguides (sometimes called “street canyons” in the literature). Systematic unaligned sampling can be a good compromise between SRS and systematic sampling as it is more robust against alignment bias, but guarantees an even distribution of sample points within the test area. Stratified sampling is typically used when there are differences and/or differences in variability in different areas. For instance, a municipality may wish to prioritize or set different performance and coverage criterion for different areas of a city.

For the purposes of geostatistical modeling (and Kriging), there are two important criteria that must be considered when selecting an intial sampling design. First, samples must cover the area to be sampled such that no two points are too far apart, which decreases interpolation resolution. And second, some number of samples must be taken at a variety of lags so that the variogram can be sufficiently estimated. In particular, clustered measurements are generally required to model small scale effects (i.e., variance from measurements separated by distances smaller than the lag distance). In [158], Olea investigates multiple

initial sampling schemes. In his approach, universal Kriging is used to select between several specific designs so that standard error is minimized. Olea strongly endorses stratified random sampling in this work, but it is not clear how well this mechanism works in other domains. In [245], Yfantis *et al.* study the efficiency of Kriging estimation for various types of sampling lattices. They find that, for the majority of cases, where the nugget effect is small relative to the variance, a triangular grid-based sample is the most efficient initial sampling scheme. In cases where the nugget variance is large and the linear sampling density is > 0.85 times the range, a hexagonal design is most efficient. The authors suggest that a small pilot sample be used to determine the empirical variogram, which can then chose an appropriate-density and grid-pattern sampling scheme for the initial sampling.

In addition to these works, if something is known about the underlying process and its variability, an optimization scheme can be used to select the best initial sample. For instance, in [230], van Groenigan et al., present a framework for Spatial Simulated Annealing (SSA) which uses a fitness function that either spreads points maximally, or chooses their lags according to a prescribed distribution. In SSA, points are varied randomly in a hill-climbing fashion so that an (at least locally) optimal sample is chosen. Additionally, if the variogram shape is known *a priori*, or a distribution of reasonable variogram parameters can be defined, then an initial sample can be chosen using SSA so as to minimize the summed or average point Kriging variance. Although related, this approach to optimizing the initial sample differs from the approach detailed in chapter 8, which seeks to optimize second-phase samples.

5.3 Interpolation

The question of interpolation is at the center of any measurement-based approach to coverage mapping. If measurements are collected at some number of points in a given region, what can be said about the points that have not been measured? Interpolation addresses this problem.

Besides general purpose spatial interpolation, there have been several papers that have attempted to develop interpolation strategies appropriate for wireless coverage mapping. In [52], Connelly *et al.* suggest a way to interpolate signal strength between RSS measurements using inverse distance weighting and claim less than 1 dB interpolation error. However, their minimal attempt to validate this, along with a lack of

realism in some of their assumptions (for instance, assuming propagation *stops* at 100 m), leaves one without much confidence in their method. In [56], Dall'Anese suggests a way to use distributed measurements from sensors to determine a sparsity-promoting WLS interpolated coverage map. This work is in the space of cognitive radios, so the authors assume that the location of sensors is not controllable and that the principle application is in empirically determining a safe transmit power for a given radio so as to avoid interfering with Primary User (PU)s. In [119], Konak proposes the use of ordinary Kriging over grid-sampled data for mapping coverage and shows that this approach can outperform a neural-network trained model presented in [150]. Finally, [161] provides a tutorial addressing the use of basic geostatistical interpolation for estimating radio-electric exposure levels. While not strictly the same as wireless network propagation, the approach is certainly relevant.

In addition to these works, there have been several recent publications by Riihijärvi and colleagues that discuss the use of spatial statistics to model radio propagation [195, 236]. As with [56], Riihijärvi's work is in the cognitive radio space, where the goal is to determine the signal at a given point from a PU so that a secondary user can choose when and where it is safe to transmit without interfering. Like [119], this work presumes a regular grid-based sample. Measurements are used to fit a semivariogram and several underlying functions are investigated. In [193], the authors suggest how this method can be used to more compactly store radio environment maps and in [194] the same authors look at how the placement of transmitters, terrain roughness, and assumed path loss effects the efficacy of the interpolated field. The theoretical work by Riihijärvi here is solid and is very inspiring, yet has two important limitations that this chapter (and the following) aims to address: (1) Riihijärvi does not evaluate the model with real measurements and hence it is difficult to say how well this approach would work in practice and (2) the work does not concern itself with where measurements are made and assumes simple grid-based sampling for measurement. The work of this thesis will build upon the work of Riihijärvi by making an empirical evaluation of these geostatistical techniques, applying them to the general case of coverage mapping, and exploring more advanced (optimized) sampling strategies.

As compared to alternative methods of interpolative mapping such as Inverse Distance Weighting (IDW), Kriging has three important benefits: (1) it is preceded by an analysis of the spatial structure of the

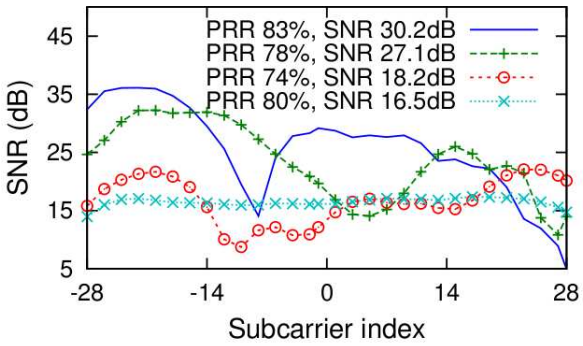


Figure 2: Channel gains on four links that perform about equally well at 52 Mbps. The more faded links require larger RSSIs (i.e., more transmit power) to achieve similar PRRs.

Figure 5.3: Example of frequency selective fading. Figure taken from [90].

data and thus an estimate of the average spatial variability of the data is integrated into the interpolation process *vis a vis* the variogram model, (2) it is an exact interpolation method, meaning that when data is available at a given point, the interpolated map has exactly that measured value at that point; and (3) since it is a robust statistical method, it provides a per-prediction indication of estimation standard error via the square root of the Kriging variance [233].

5.4 Method

This section describes the method for geostatistical mapping developed for the radio environment in this thesis. Of course, the method described here was not found on first trial and is the result of many false starts, mis-turns, and exciting discoveries. Although the description here is on the final method developed, some time will be given to explaining why particular design choices were made. The first subsection begins by discussing the important question of performance metrics and measurement, and the following sections will discuss the process of geostatistical mapping, which involves sampling design, spatial structure analysis, de-trending, variogram fitting, Kriging, and finally mapping and basic visualization.

5.4.1 Performance Metrics and Measurement

The question of what to measure or, which metric is the best predictor of network performance at a given point, is intimately tied to the efficacy of any coverage mapping system. The vast majority of research

has been done on measuring and predicting signal, noise, and SNR. However, there has been a vocal minority of researchers and network operators that have been claiming for some time that single value SNR is a poor predictor of network performance. In [132], Lee *et al.* propose a method for averaging signal strength measurements over a region of 20 to 40 wavelengths in order to average out small scale fading effects and obtain a more stable indicator of signal strength. By and large, this averaged SNR metric is what is used by cell network operators today.

5.4.1.1 Metrics for Simultaneous Multiple Streams

More recently, there has been some work to develop metrics appropriate for wide-band Orthogonal Frequency Division Multiplexing (OFDM) waveforms and Multiple Input Multiple Output (MIMO) streams. In [92], He and Torkelson present an "effective SNR" metric for OFDM systems that involves averaging and combining across subcarriers. The most recent paper in this thread is [90] by Halperin *et al.* In this work, the authors use 802.11n radios with a customized firmware to collect the Channel State Matrix (CSM), which contains SNR fading values for each subcarrier. This allows them to develop a practical metric of performance that takes into account frequency-selective fading effects, where some OFDM subcarriers are attenuated more significantly than others. Figure 5.3 gives an example of this phenomenon. The authors propose a metric called "effective SNR", which is the SNR value that would acheive the same bitrate for the given modulation scheme as the average bitrate across all the subcarriers:

$$\hat{r} = 1/N \sum_{i} BER_k(SNR_i) \tag{5.10}$$

$$SNR' = BER_k^{-1}(\hat{r}) \tag{5.11}$$

where SNR_i is the SNR of the i^{th} subcarrier, N is the number of subcarriers (52 for 802.11 OFDM implementations), $BER_k()$ is a function that computes the expected Bit Error Rate (BER) for a given modulation scheme k and $BER_k^{-1}()$ is the inverse function that produces the SNR required to achieve a given BER, $\hat{r}$ is the average BER across all subcarriers, and SNR' is the computed effective SNR. In their paper, the authors

show that this metric is better able to predict the performance of real wireless links than average (across the 10 MHz channel) SNR.

5.4.1.2 Higher-Layer Metrics

As is discussed in [92], BER is the ideal metric of performance for a given technology and propagation environment. However, in practice its collection is costly. In chapter 4, a state-based metric of performance that incorporates tests at multiple layers was discussed. In situations where the technology is fixed, application-layer tests are obviously the best metric of performance. However, in scenarios where measurements are being made in order to predict the propagation environment alone and should not be tied to a particular technology, modulation, waveform, or rate adaptation scheme, lower-level metrics are most useful (for instance, in cognitive Radio Environment Map (REM) applications). For this reason, to maintain generality this thesis will focus on lower-level metrics such as SNR and effective SNR, while performing due diligence to understand how and how well these metrics correlate with higher-layer metrics in each of the environments studied.

There have been some recent proposals which hint at a bountiful future for robust low-level metrics that can be collected with commodity hardware. For instance, in [22], Firooz *et al.* propose a way to use the GNU Radio Software Defined Radio (SDR) platform [179] to implement a Channel Impulse Response (CIR) metric based on an 802.11 transceiver, which can be used to measure delay spread. A practical method for measuring delay spread could be used to model multipath fading and ISI effects with high precision. Another promising example is [184], where Rayanchu *et al.* show that Commercial Off The Shelf Equipment (COTSE) hardware can be used to measure and model power from interfering stations and devices. Because the methods described here are agnostic to the underlying metric used, the mapping approach can be upgraded simultaneously as better methods are developed to estimate channel performance.

5.4.1.3 Estimating Channel Occupancy

Although the case studies in this thesis focus on the task of understanding the performance of a deployed network, it should be noted that the same techniques could be used to map an interfering network

or discover empty spectrum for Cognitive Radio (CR) applications. In [23], Anderson and Cameron discuss results of a spectrum survey around Annapolis, Maryland. They find that estimations of channel occupancy can vary substantially depending on how quickly the spectrum is scanned. In particular, fast scans tend to overestimate channel occupancy because they are effected by transient signals, while slow scans tend to underestimate occupancy. The authors also suggest that because some narrowband technologies operate at very low SNR values, highly sensitive receivers are required to detect low-power transmissions. Some exciting recent work by Rayanchu *et al.* has shown that it may be possible to collect sufficiently accurate information about cross and interband interference using COTSE commodity hardware [184].

5.4.2 Selecting a Sampling Density and Pattern

As was discussed in section 5.2, the spatial sampling literature suggests that an equilateral triangular uniform lattice is often the most efficient sampling strategy for two-dimensional spatial processes, and thus this strategy is utilized in this work. A uniform equilateral triangular lattice of a given lag h in meters is generated as described in algorithm 2. Choosing a sampling density, or lag h, requires consideration of a number of tradeoffs. Firstly, there are fundamental limits in terms of the smallest meaningful lag. In [208], Shin et al., suggest that there is substantial spatial autocorrelation of measurements (of IEEE 802.11 networks) within 1m. In [132], Lee *et al.* suggest averaging measurements within 20 to 40 wavelengths to avoid overfitting a model on the noise from fast fading. Finally, because a typical GPS unit has a working accuracy of between 1.5 and 10m (depending on the environment and weather), there is little benefit to sampling at a greater density than this, since the subsequent model fitting would be, in effect, only fitting noise from the locationing error. This distance works out to between 4.8 and 2.4m at 2.5 GHz. Hence, any measurement-based interpolation should not expect to produce a map with finer resolution than 20 to 40 wavelengths, in a scenario with fine locationing resolution, and no smaller than ≈ 5m in a scenario using commercial GPS for positioning.

Based on this reasoning, one might endeavor to take as many measurements as possible at a lag slightly above this lower bound. However, in many situations, measurements are costly to collect in terms of both time and money. It is one of the aims of this thesis to provide a sampling methodology that is relatively

Algorithm 2 Compute Equalateral Triangular Sample in 2-Space

1: $h \leftarrow$ desired lag in meters
2: $R \leftarrow$ the radius of the earth in meters
3: $step \leftarrow (h/R) * (180/\pi)$
4: $lngmin \leftarrow$ minimum longitude of bounding box
5: $lngmax \leftarrow$ maximum longitude of bounding box
6: $latmin \leftarrow$ minimum latitude of bounding box
7: $latmax \leftarrow$ maximum latitude of bounding box
8: $h \leftarrow latmax$
9: $w \leftarrow lngmin$
10: $nh \leftarrow 0$
11: $nw \leftarrow 0$
12: **while** $h > latmin$ **do**
13: **while** $w < lngmax$ **do**
14: $nw \leftarrow nw + 1$
15: $w \leftarrow w + step$
16: **end while**
17: $nw \leftarrow 0$
18: $w \leftarrow lngmin + step/2$ if nh is odd, otherwise $lngmin$
19: $h \leftarrow h - step$
20: $nh \leftarrow nh + 1$
21: **end while**

minimal in terms of the work required. For this reason, it is most desirable to select a sampling density that provides enough information to sufficiently model the important details of the network coverage, without requiring more work than is necessary. In order to understand how sampling density affects the ability to perform a meaningful fit, an experiment was performed using the data from the "pdx" drive-test described in 4.2.1.

In this experiment, grids are generated at multiple lag distances. For each point on the grid sample, the nearest measurement point is located for each of the 72 APs using a nearest neighbor algorithm. Sample points that are not within h meters of any measurement are discarded. To manage this task efficiently, measurements are inserted into a PostgreSQL database with PostGIS extensions [160, 88]. Figure 5.4 shows a portion of the sample, with those sample points that are within 40 wavelengths of a measurement highlighted.

Figure 5.5 shows the original measurements, as well as resampled measurements with varying lag distances for a representative AP called "pdx90". These figures plot the path loss in dB (calcuated by solving equation 2.1 for PL). From the top figure, 5.5a, which shows the raw measurements, it is clear to see the measurement bias due to drive-testing—all measurements fall in straight lines confined to streets. Each sample at a lesser resolution (larger lag) approximates the original data with decreasing fidelity. This can be seen clearly by inspecting both the spatial distribution and the shape of the value distribution (which appears to be lognormal, as we might expect). Although all four resamplings capture the basic value distribution, there is a clear loss of information when the lag is greater than 100 m.

Figure 5.6 gives another view of this data that is more common in the propagation modeling literature. Here, the observed relationship between path loss and distance is plotted for each sample. Each plot also provides a linear least squares regression fit to the data using the modified version of equation 2.8: $PL = 10\alpha log_{10})(d) + 20log_{10}(f) + 32.45 + \epsilon$ where α and ϵ are the fitted slope and intercept and correspond to the path loss exponent and offset. The residual error of the fit, which can be thought of as the variability due to fast and slow fades, is given as σ. The raw measurements show an α of approximately 2.2, an ϵ of 22.8 and a σ of approximately 8.5 dB. These are all reasonable and expected values for outdoor radio transmissions. As before, each successively sparse sample can be viewed as a reduced fidelity approximation of the underlying

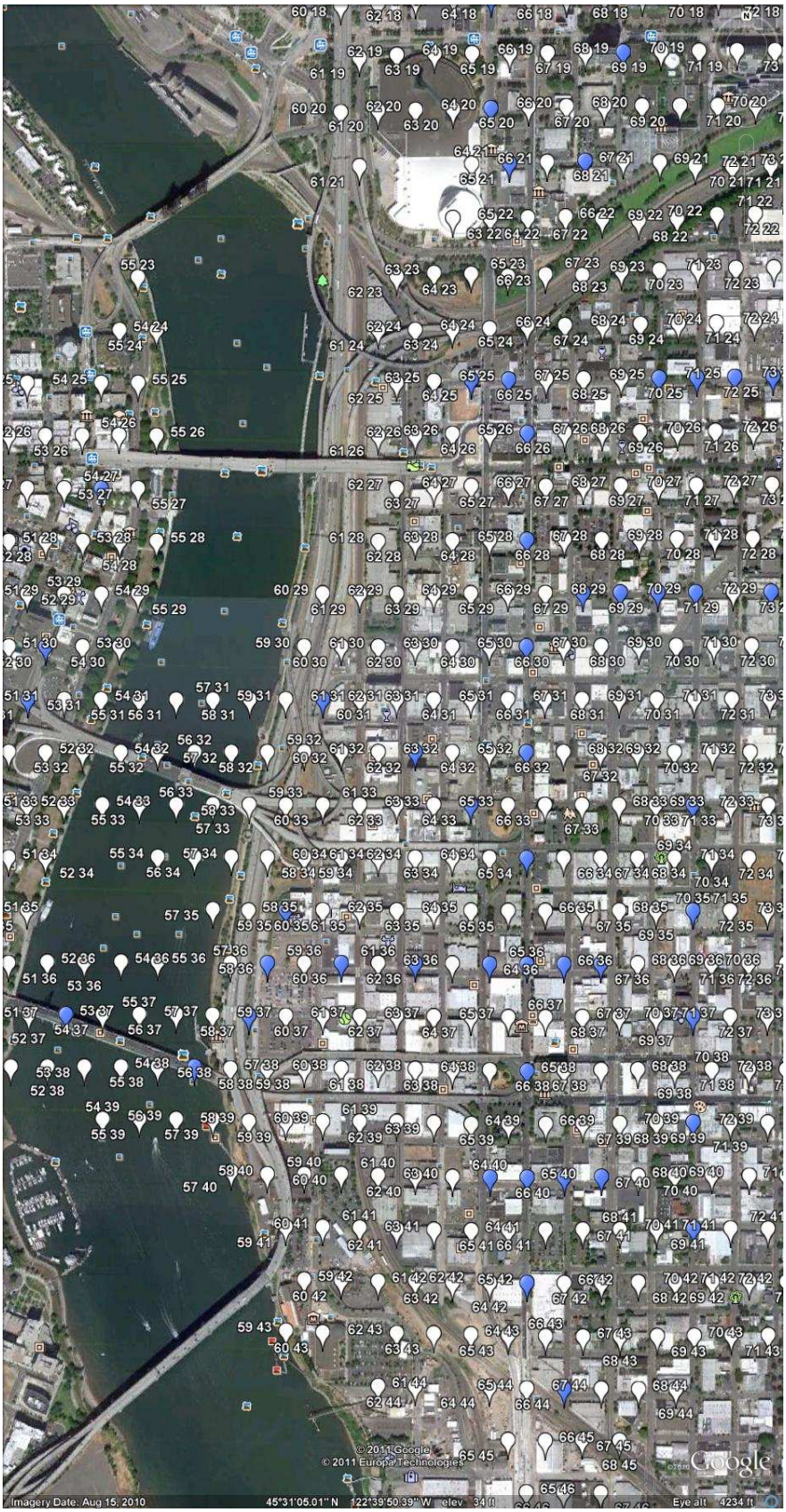


Figure 5.4: Example of uniform equilateral triangular sample with $h = 100m$. Markers highlighted blue (as opposed to white) are within 40 wavelengths (approximately 5 meters) of a measurement point.

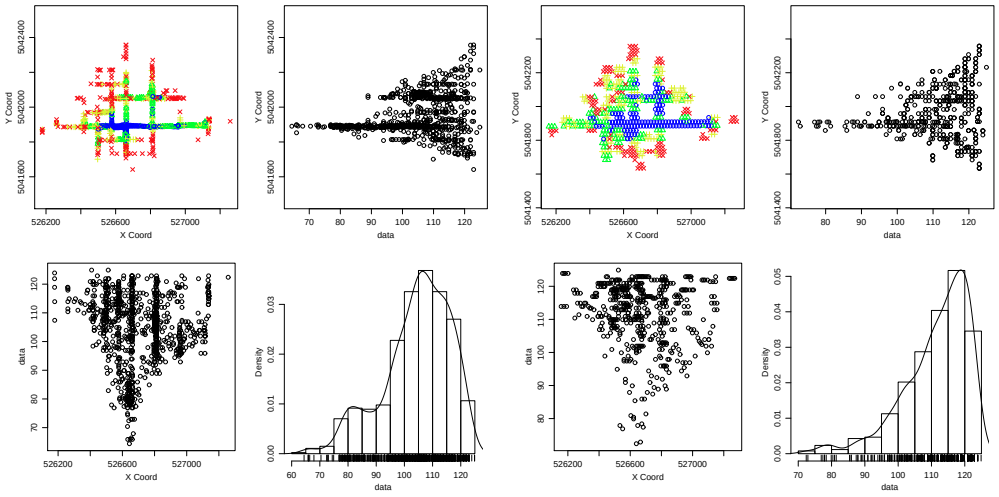
relationship and variability. At $h = 50m$ and $h = 100m$, the plots are very close. However as the lag is increased to 250 and 500 m, there are no longer sufficient samples to create an accurate picture. As a result, the fitted α, ϵ, and σ values radically underestimate those from the original measurements. Given this, a density of 100 m or less seems to sufficiently model the basic distance-attenuation relationship of this particular data set, and larger lag distances might radically underestimate the actual attentuation.

The next task is to attempt to characterize the semivariance of the measurements with a semivariogram model as described in section 5.1.2. Figure 5.7 shows the empirical variograms for the raw and resampled data. These semivariogram plots provide a final comparison of resampling densities. The raw data produces a semivariogram with nugget effect of approximately 25 dB (i.e., the y-axis crossing), range of 200 m (the location of the first peak) and sill of approximately 165 dB (the horizontal asymptote around the peak, which does not actually asymptote in this example). As before, the samplings less than and equal to $h = 100$ seem to reasonably well approximate this curve, while the other samplings deviate wildly. From the perspective of Kriging, none of these variograms are particularly well behaved as they do not have the characteristic shape and horizontal asymptote required of permissible semivariogram models; this will be addressed with more sophisticated fitting extensions below. Based on the results of this experiment, $h = 100$ appears to be a reasonable starting sampling density in the first case study. The process (and possible harm associated with) resampling of a biased sample to derive a uniform sample is the topic of chapter 7.

5.4.3 Krige the Residual

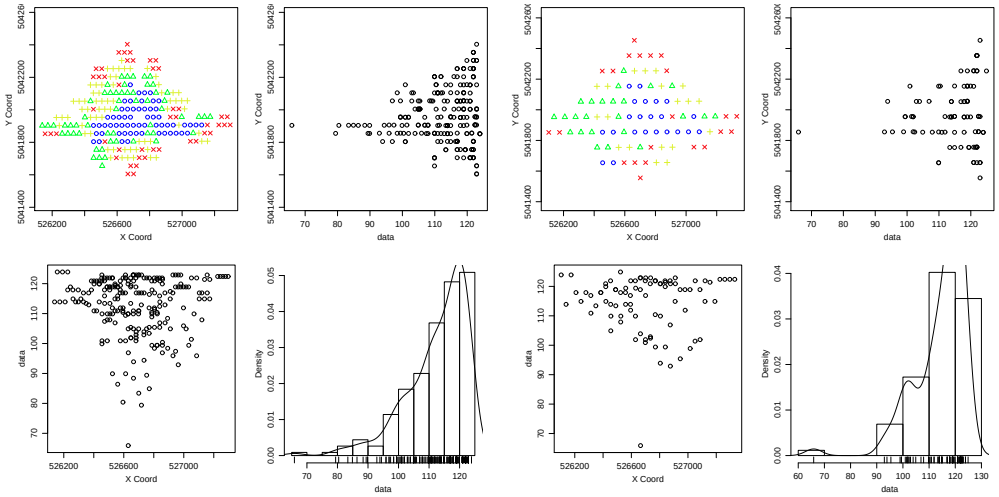
Assuming an initial sample has been obtained by either grid, random sampling, or resampling, the next task is to "de-trend" the data. In [159], Olea et al describe the importance of removing any sources of nonlinear trend from measurements so that the fitted (interpolated) field complies with the basic tenets of geostatistics. To this end, a hybrid approach is developed: a predictive (empirical) model is used to calculate the predicted path loss value at each measurement point. This prediction is subtracted from the actually observed value to obtain the residual, or excess path loss relative to the model predictions:

$$Z'(\mathbf{x}) = Z(\mathbf{x}) - P(\mathbf{x}) \tag{5.12}$$



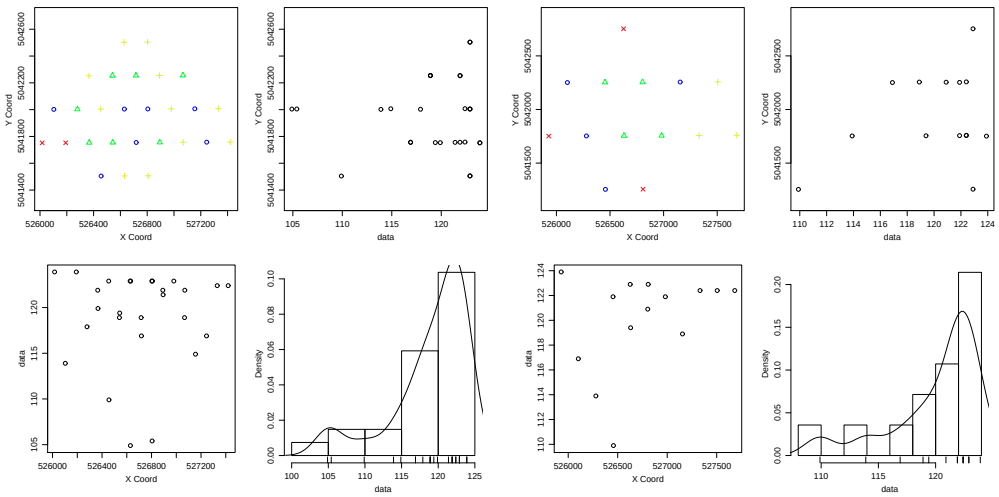
(a) Original Measurements

(b) h = 25 m



(c) h = 50 m

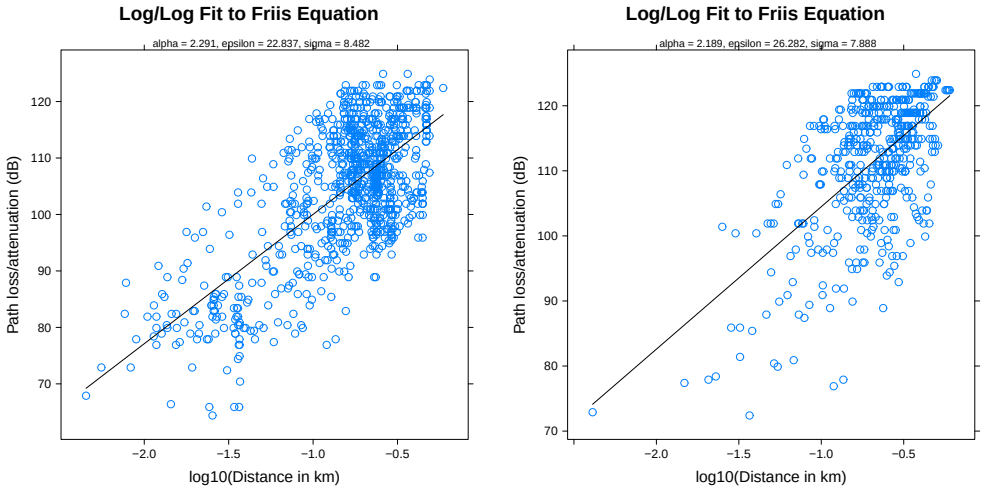
(d) h = 100 m



(e) h = 250 m

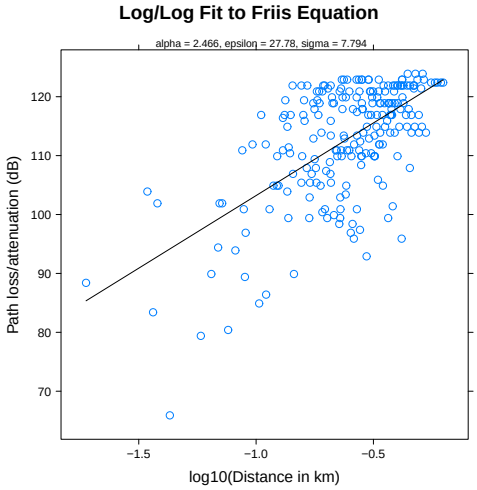
(f) h = 500 m

Figure 5.5: Path loss measurements for “pdx90” AP, both as-collected and resampled at varying lag distances. Each figure contains four plots which show the spatial and value distribution of the processes

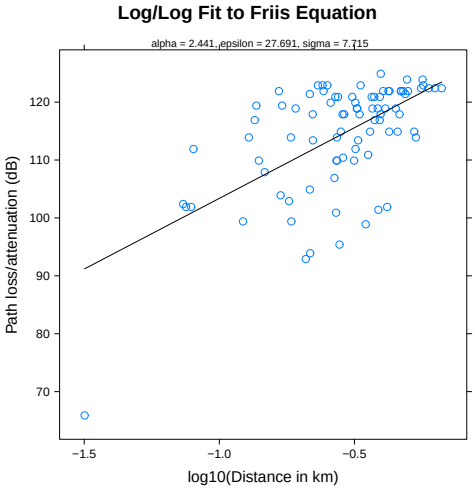


(a) Original Measurements

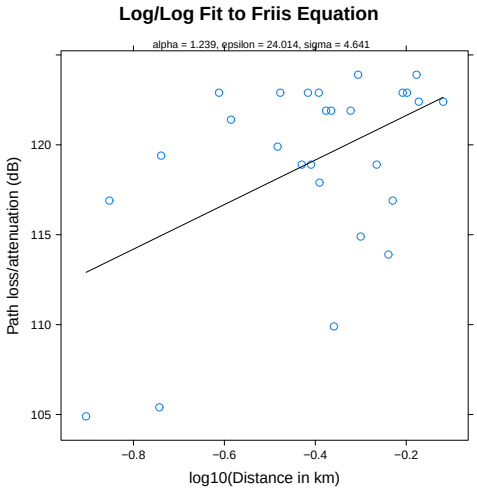
(b) h = 25 m



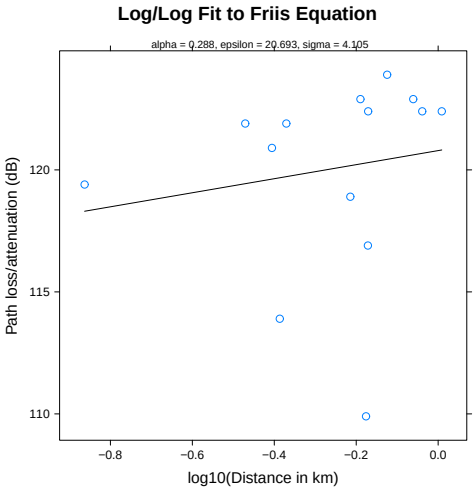
(c) h = 50 m



(d) h = 100 m

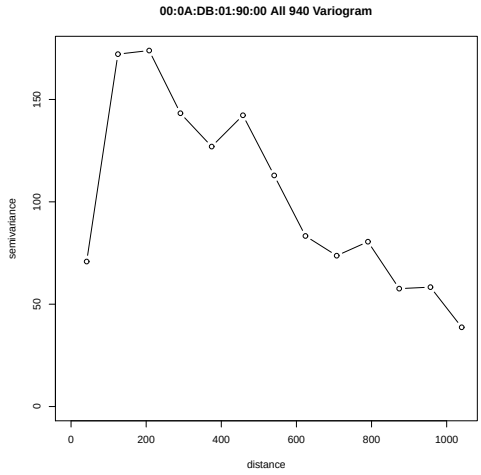


(e) h = 250 m

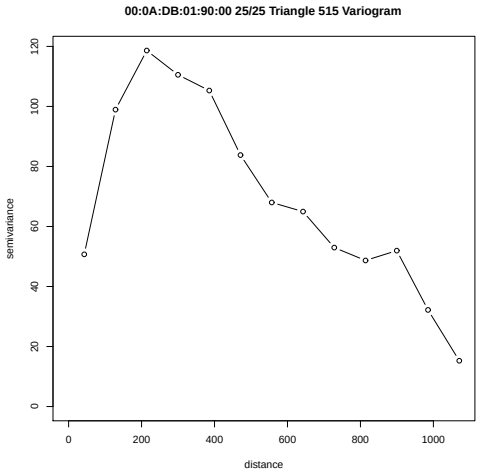


(f) h = 500 m

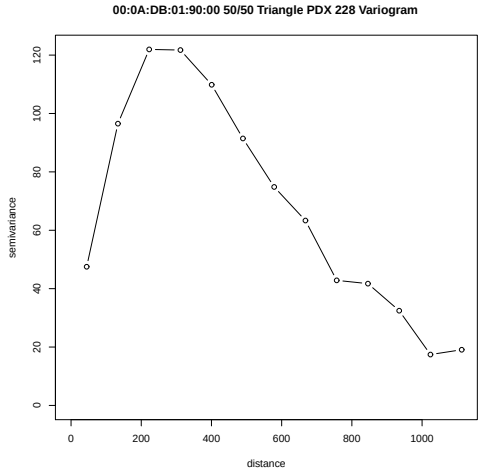
Figure 5.6: Path loss measurements for “pdx90” AP, both as-collected, and resampled at varying lag distances. Each figure plots the path loss as a function of distance on a log-log plot. A linear least squares regression fit line, and parameters are given.



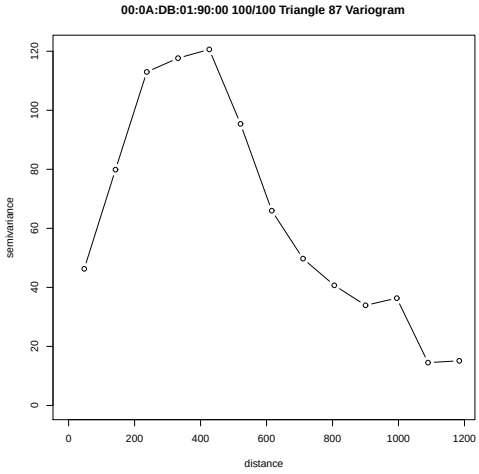
(a) Original Measurements



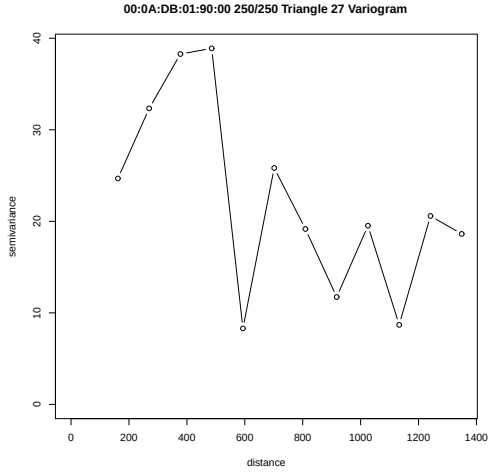
(b) h = 25 m



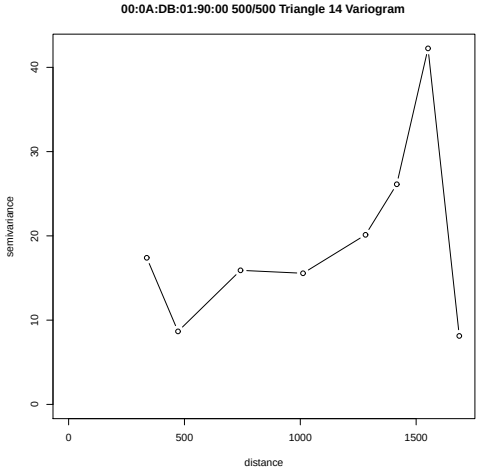
(c) h = 50 m



(d) h = 100 m



(e) h = 250 m



(f) h = 500 m

Figure 5.7: Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances.

where $Z'()$ is the residual (de-trended measurements) process, $Z()$ is the observed process and $P()$ is the model prediction. Chapter 3 showed that the best-case performance of *a priori* models, when tuned to their ideal parameters, was on the same scale as the residual error of a log-log fit to path loss using a small number of measurements. Given this, an empirical approach to modeling seems easy to advocate here. For the measurements from each access point, fitted α and ϵ values are determined so that the trend can be removed as follows:

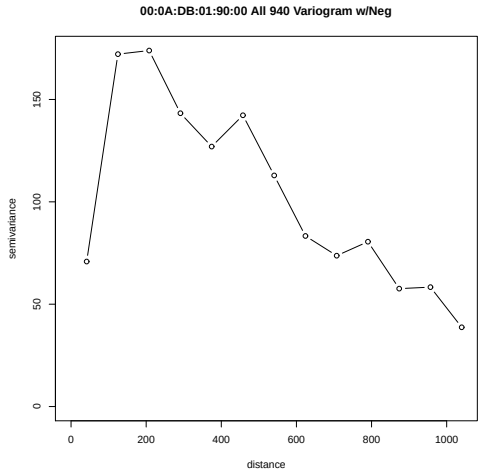
$$Z'(\mathbf{x}) = Z(\mathbf{x}) - (\alpha 10log)10(d) + 20log_{10}(f) + 32.45 + \epsilon) \tag{5.13}$$

the resulting de-trended observations can then be used to fit an empirical variogram as described in section 5.1.2. This is a pleasing approach to de-trending because it is entirely modular and extendable—the fitted log-log empirical model described here can be easily replaced with any other predictive model. In this way, the interpolation process can be viewed as careful way to correct for any remaining (environment-specific) model error, instead of as a complete replacement. As the state of the art in path loss modeling is advanced further, and models are able to make predictions closer to measurements, this improvement can be carried through to measurement-based interpolation in the process of de-trending as described here.

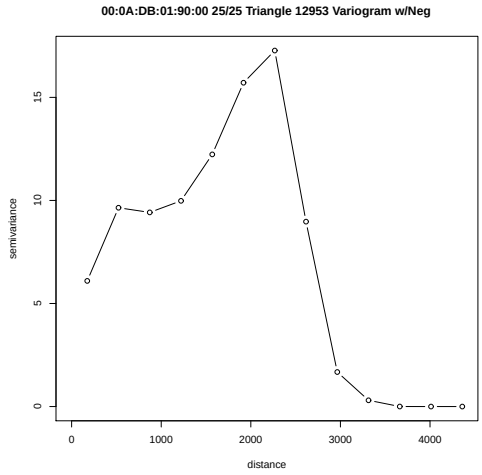
5.4.4 Variogram Fitting

Figures 5.8 and 5.9 show the resulting empirical variograms before and after de-trending the data for AP pdx90. The nugget tolerance (i.e., points within this distance are considered co-located) is set to 40λ per the classic averaging recommendations of Lee in [132]. As before, at larger lag distances the plotted variograms deviate wildly from ground truth. However, even using all measurements or a dense ($\leq 100m$) sampling, the variogram does not permit fitting with any of the permissible models described in [112]. In order to tame the variogram model, two more changes are necessary.

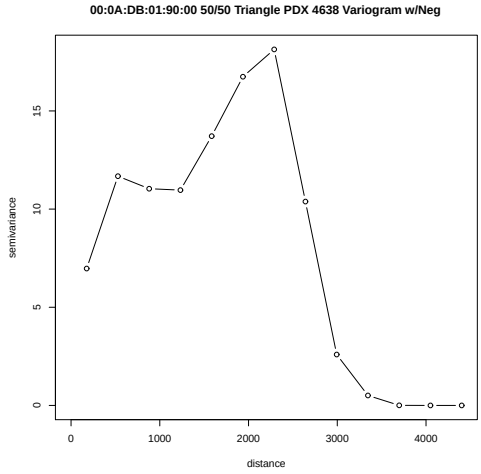
First, “negative results” must be included in the dataset: when a given point has been visited, and the instrument failed to observe a given AP at that point, it can then be inferred that the signal at that point must be lower than our receiver threshold. To assuage this, at points visited where there is no data for a given AP, a measurement with an unrealistically low value is used. The approximate noise floor (-95 dBm for most



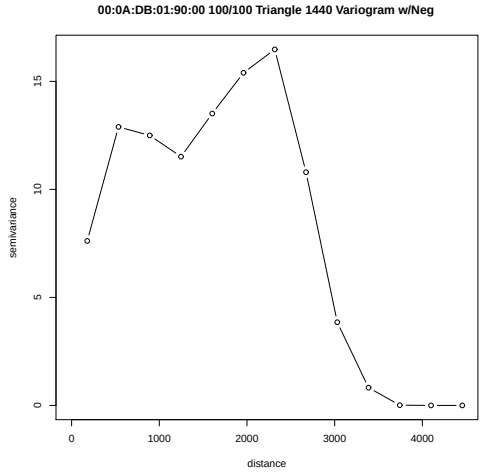
(a) Original Measurements



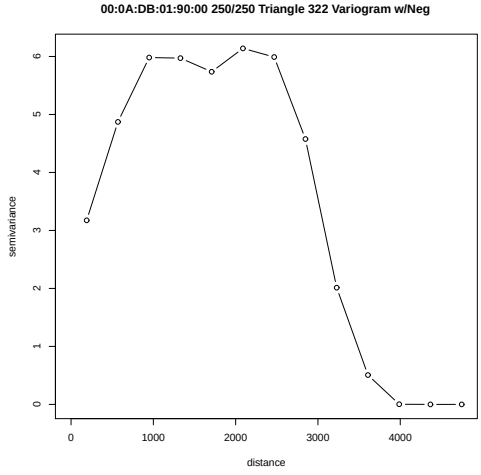
(b) h = 25 m



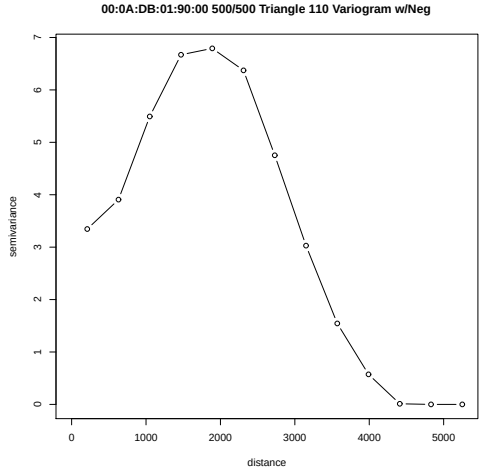
(c) h = 50 m



(d) h = 100 m

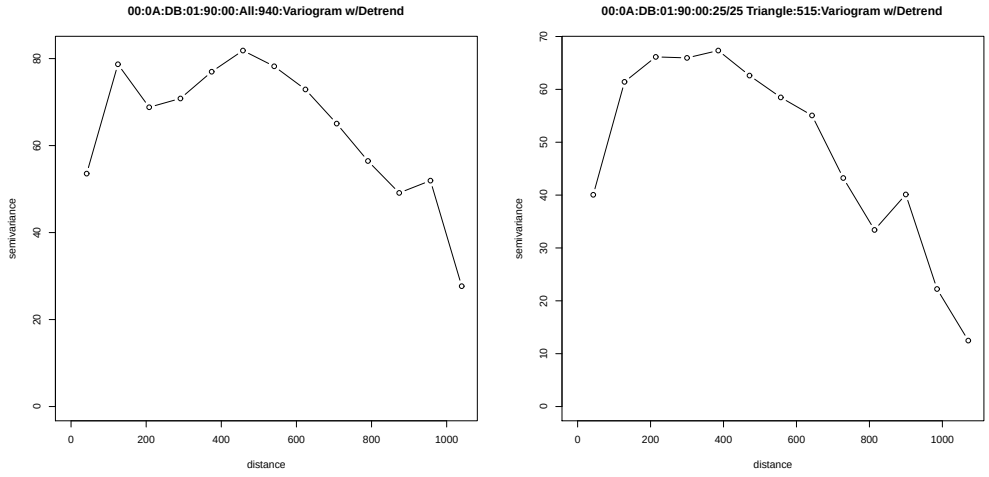


(e) h = 250 m



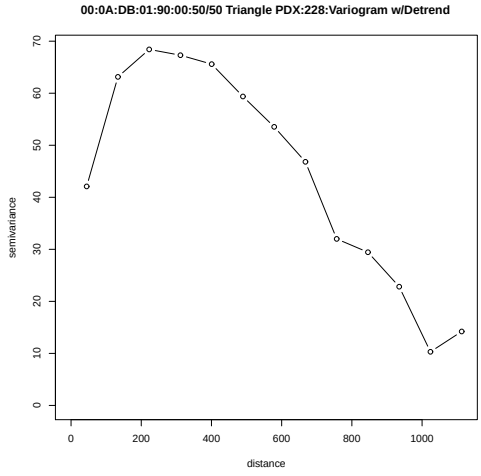
(f) h = 500 m

Figure 5.8: Empirical semivariograms of path loss for “pdx90” AP without de-trending.

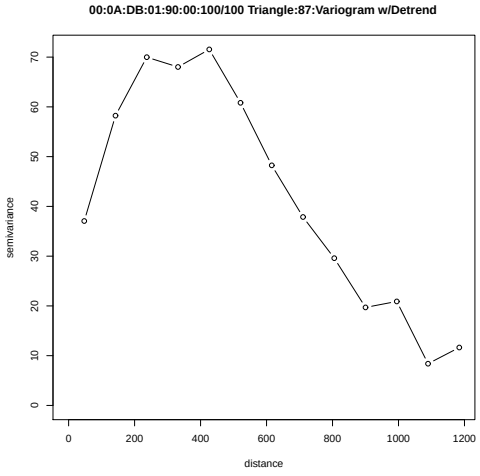


(a) Original Measurements

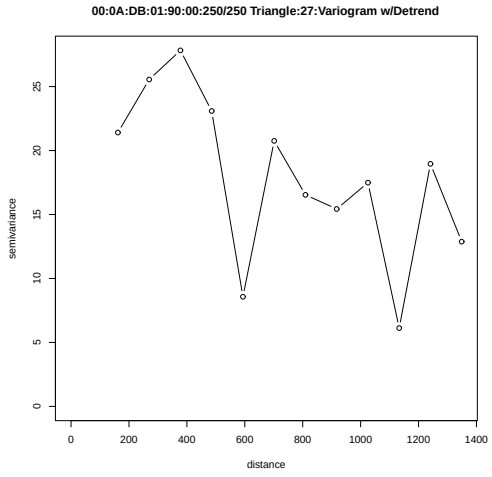
(b) h = 25 m



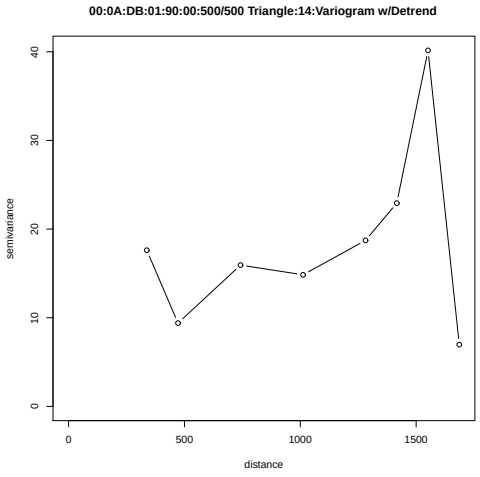
(c) h = 50 m



(d) h = 100 m



(e) h = 250 m



(f) h = 500 m

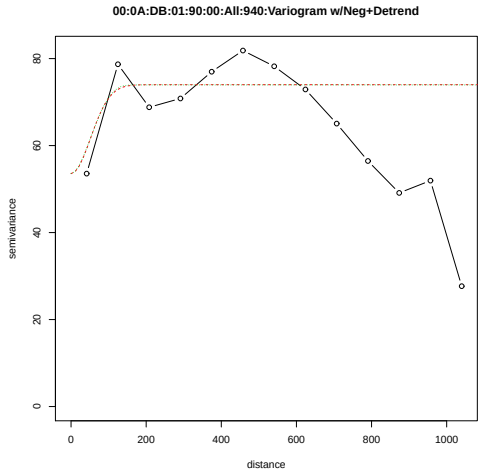
Figure 5.9: Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with Frii’s freespace model used for de-trending.

transcievers at the frequency we're operating on) is a reasonable choice. This is an important extension: if a point has been visited, the lack of observation of a given AP carries information in itself. If no measurement is record at this point, there is substantial information loss. Instead, by recording a measurement at a very low value, the way the observed signal tapers towards the noise floor near the edge of each AP's coverage can be modeled explicitly. Figure 5.10 shows the resulting empirical semivariograms after this modification. These plots also show fit lines for Gaussian and cubic semivariogram models, which are poor.

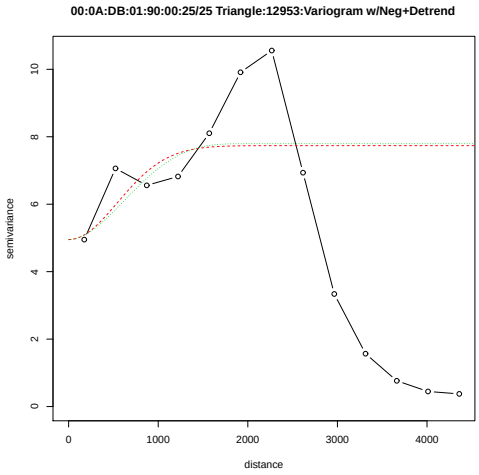
In large part the fits are poor because the empirical variogram models exhibit a large hump and then trail off afterward. This may be because observations at large lag distances are scarce because it is simply unlikely to observe a transmission further than say, 2 km, apart in a network of this kind. Hence, fits are truncated near the peak of the models and instead focus on fitting the portion of the curves where data is available and allow the remainder to lie under the asymptote. For this data set, several options were explored for truncation and it was found that truncating at $d = 800m$ worked well for all APs and nearly all resampling densities. Figure 5.11 shows the resulting empirical semivariograms and their fits with the cubic and Gaussian models. In general, cubic and Gaussian models fit well for nearly all datasets studied in this thesis. This is a pleasing result, since the cubic model is closely related to the spherical model, and the Gaussian model is closely related to the exponential model, both of which have been suggested as a reasonable choice in modeling the semivariograms of electromagnetic processes [119, 236]. Variogram fitting is accomplished with weighted least squares as described in [112] using the implementation available in the R package "geoR" [114].

5.4.5 Goodness of Fit

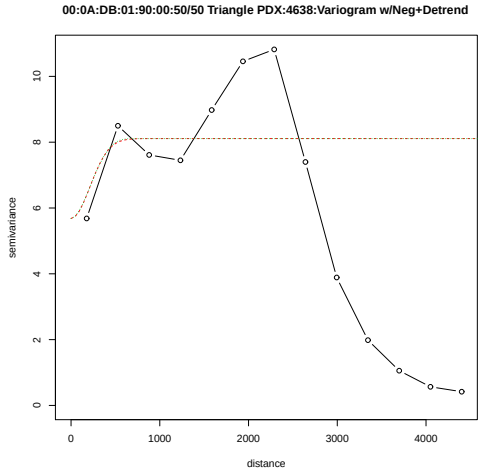
In the implementation used in the case studies below, this fitting process is automatic: fitting is performed using both the cubic and Gaussian models, with and without distance truncation, and with and without null measurements included. Then, the goodness of each fit is calculated using k-fold cross validation (typically with $k = 10$). For each fold a random sample of 20% of measurements is excluded, a fit is made with the remaining points, and the excluded points are predicted with the fit. The resulting RMSE and Mean Square Kriging Variance (MSKV) of these predictions are calculated and averaged across the k



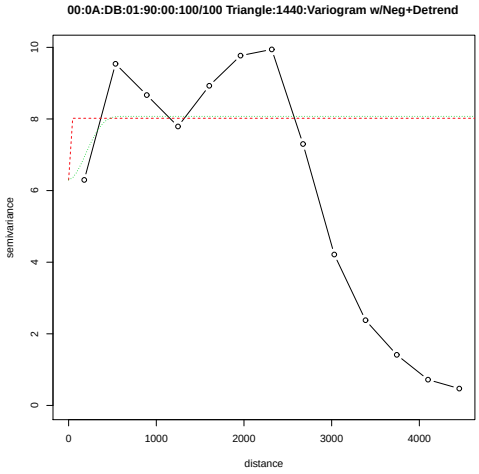
(a) Original Measurements



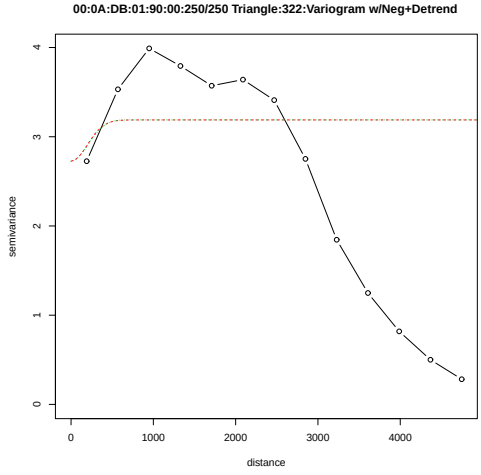
(b) h = 25 m



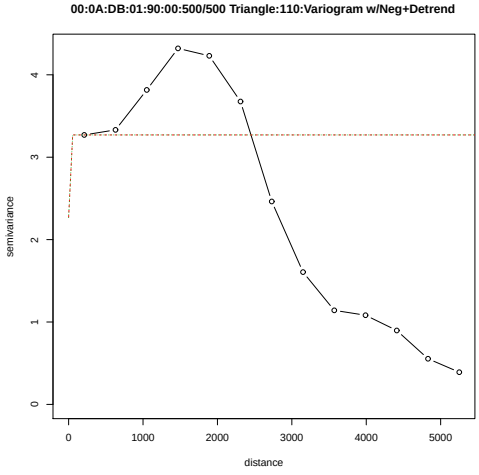
(c) h = 50 m



(d) h = 100 m



(e) h = 250 m



(f) h = 500 m

Figure 5.10: Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with offset- and slope-fitted Frii’s freespace model used for de-trending. Measurements at the noise floor have been inserted at points where an observation was unable to be made.

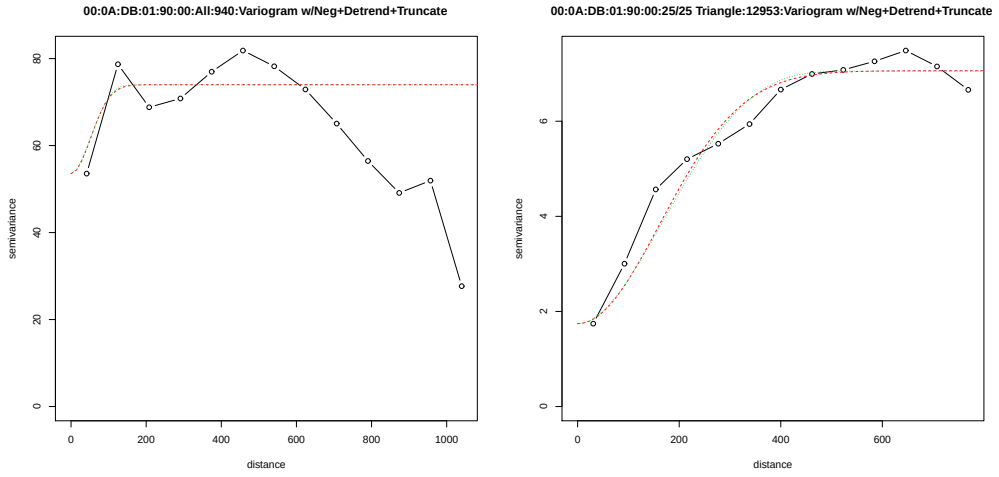
folds. The fit with the lowest mean RMSE is then used for Kriging. As a general rule of thumb, truncated fits that include null measurements are the best performing in our case-studies, so these are focussed on in the analysis. However, the split is fairly even between the Gaussian and cubic models.

5.4.6 Mapping with Ordinary Kriging

With fitted variogram models in hand, the next task is to generate a coverage map. This is a straightforward but computationally intense process that involves Kriging the value at each pixel in the mapping area. To accomplish this, Ordinary Kriging (OK) is applied as described in [119], using an implementation in the "geoR" R library [114].

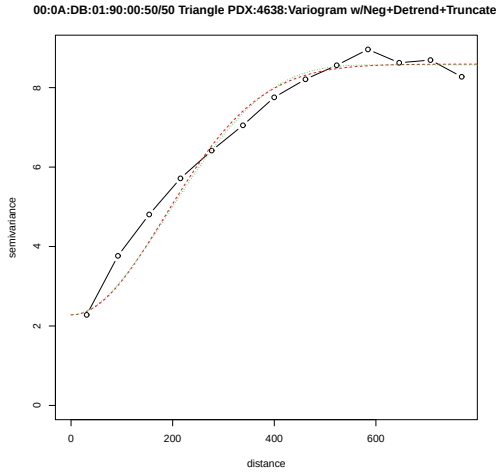
Two important questions that arise in mapping are: what resolution should be targeted, and what colormapping (visualization) strategy should be used? The computational demands of the process are intimately tied to the resolution, and maps of an arbitrarily fine resolution cannot be generated in a reasonable amount of time using basic Kriging methods (which, afterall, must solve a quadratic program of some size for each pixel!). In practice, a resolution of 0.05 Pixels per Meter (PPM) (one pixel for each 20x20m square) is good for quick estimates and a resolution of 0.2 PPM (one pixel for each 5x5m square) is the highest reasonable resolution. Because a nugget tolerance to 40 wavelengths (approximately 5m at 2.4 GHz) is used per the discussion in the previous section, it would not be meaningful to generate images at a higher resolution this. Calculating a map at the 0.2 PPM resolution can require a substantial amount of memory and fails for some maps with more data than the other resolutions. This stems from the fact that R requires its data structures to be allocated in contiguous blocks of memory and often a chunk of memory of the size required simply isn't available. For situations where a 0.2 PPM resolution map is intractable to generate, the 0.05 PPM resolution map can be used instead.

Using the de-trended measurements and fitted variogram as input, the Kriging process will produce a map matrix defining a map of excess path loss above or below the predictions of the fitted empirical model used for de-trending. These predictions are now added back to the Kriged map to create a map of the signal strength. Additionally, at each Kriged location the residual Kriging variance can be computed, which can be used to generate a map of residual uncertainty. Because Universal Transverse Mercator (UTM) coordinates

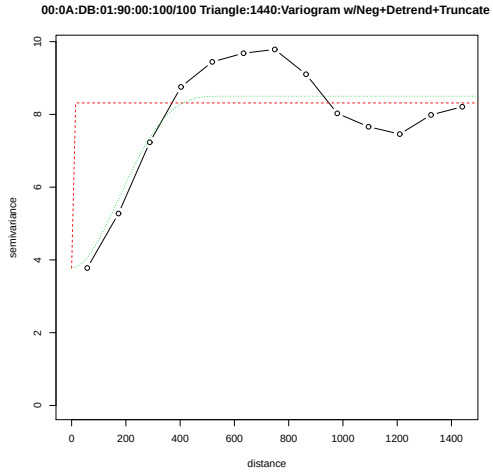


(a) Original Measurements

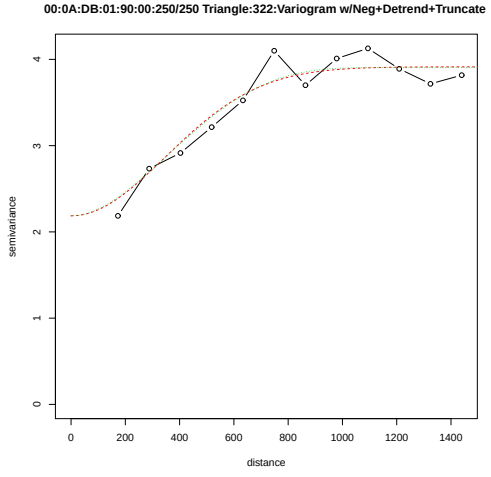
(b) h = 25 m



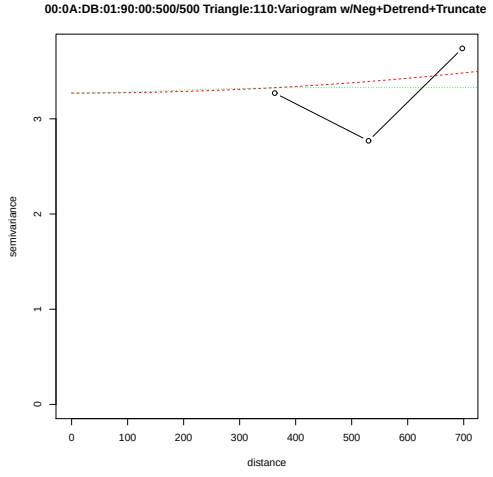
(c) h = 50 m



(d) h = 100 m



(e) h = 250 m



(f) h = 500 m

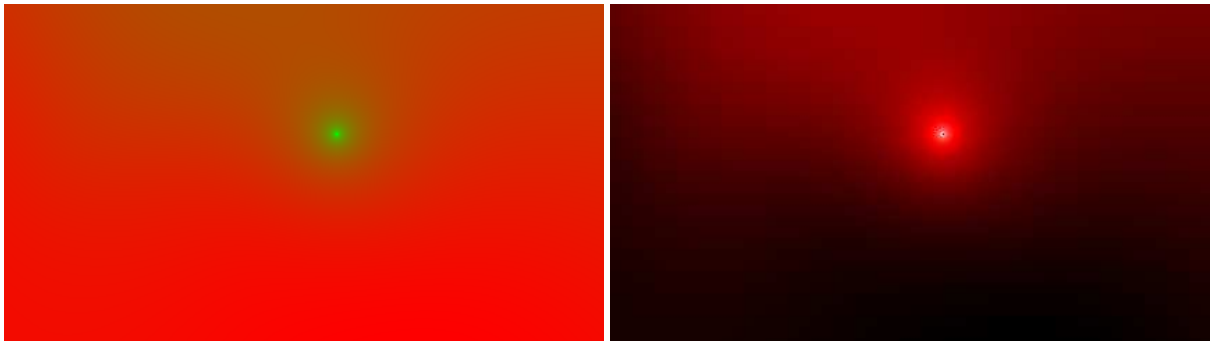
Figure 5.11: Empirical semivariograms of path loss for “pdx90” AP resampled at varying lag distances with Frii’s freespace model used for de-trending. Measurements at the noise floor have been inserted at points where an observation was unable to be made.

are used for all locations and mapping, the final map is simply a square meters-based grid with true north being upwards, as one would expect.

5.4.7 Visualization

These maps can produce different interpretations depending on the way values are combined, or even just through the color scheme. In [201], Rogowitz *et al.* show how choice of color map can dramatically effect the way data is interpreted and suggest careful choice of an appropriate color map for a given application that both varies luminance and saturation in addition to hue. Popular radio planning applications such as SPLAT! [136] and RadioMobile [54] color pixels on a map depending on the predicted SNR at that point. This color map can be overlayed on a terrain map to show their relationship (which is substantial because these tools use the ITM). In the planning tool proposed by Bartels *et al.* in [35], a logarithmic color encoding is used to show predicted coverage [152]. In [200], Robinson *et al.* take a simple domain-oriented visualization scheme and show coverage as a region of circles and coverage gaps (holes) as red x's. However, by and large there has been little substantial work studying visualization strategies for coverage maps. The maps in presented in this thesis take advantage of hue and luminance as suggested by [201], showing coverage on a color scale from (dark) black to (bright) red. By overlaying the luminance and saturation scale on the hue, contours not immediately visble in a simple hue-based (i.e., reg/green) interpolation can be readily seen. Figure 5.12 shows an example of the difference between these color maps using the example of WiMax coverage from a single BS on the CU campus.

A final example of how these maps might be used in practice is given in figures 5.13, and 5.14, which show Kriged maps here overlayed on Google Earth orthoimagery [82], Digital Terrain Model (DTM)s, and three-dimensional building models. In this example, the Kriged map is reprojected into the correct coordinate system and is placed at the correct coordinates in the Google Earth software using a Keyhole Markup Language (KML) file. An alpha channel is introduced to add partial transparency. Although a great deal of work could be devoted to domain-oriented visualization strategies for wireless coverage maps, simple interactive map overlays such as these, using widely available map tools like Google Earth, may be a strong first step in the direction of interactive mapping and planning tools for empirically derived wireless



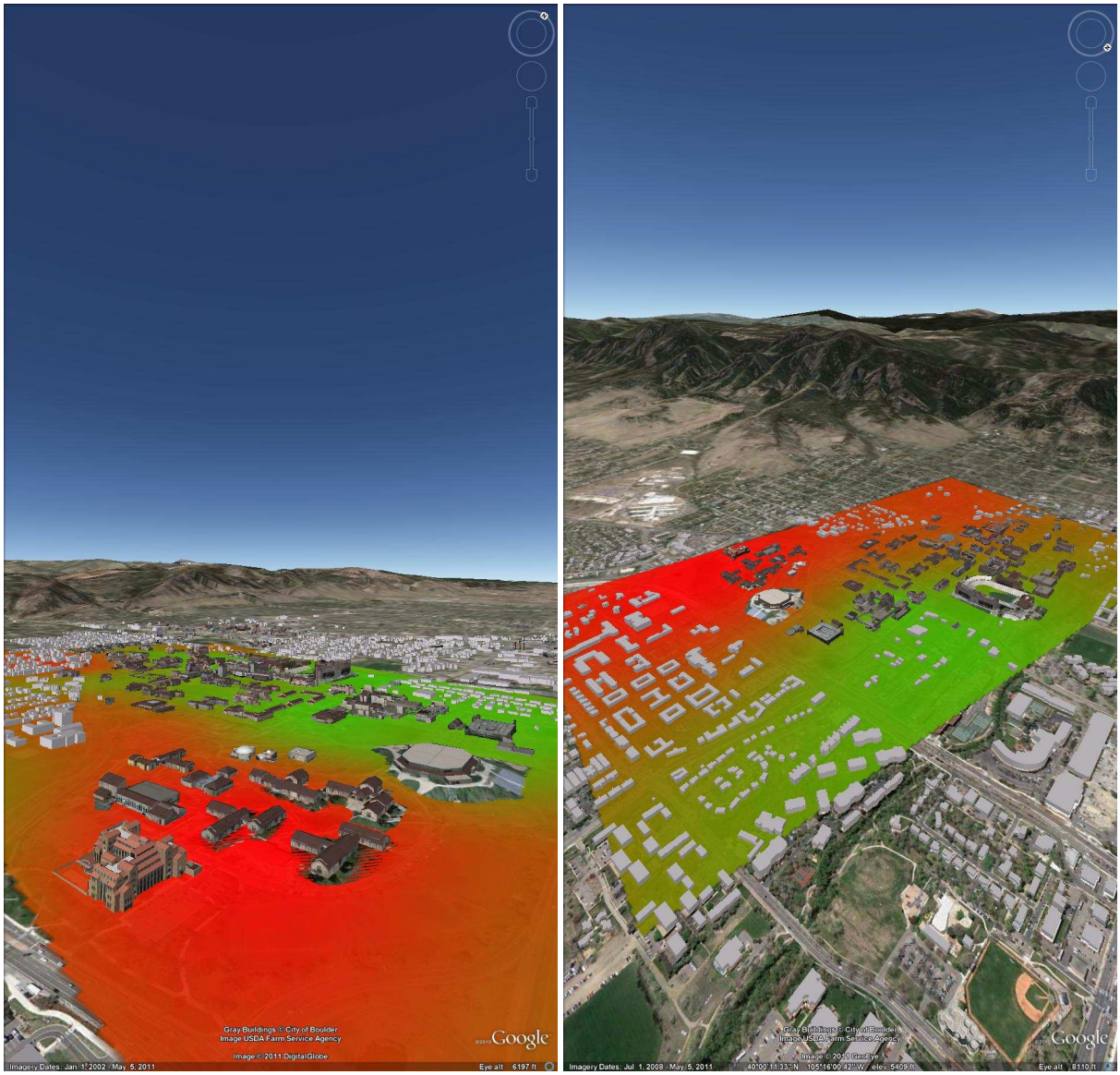
(a) Hue-Only Red/Green Map (b) Hue-Luminance Red/Black Map

Figure 5.12: Comparison of color maps using a Kriged map of WiMax CINR coverage for one BS at the University of Colorado.

coverage maps.

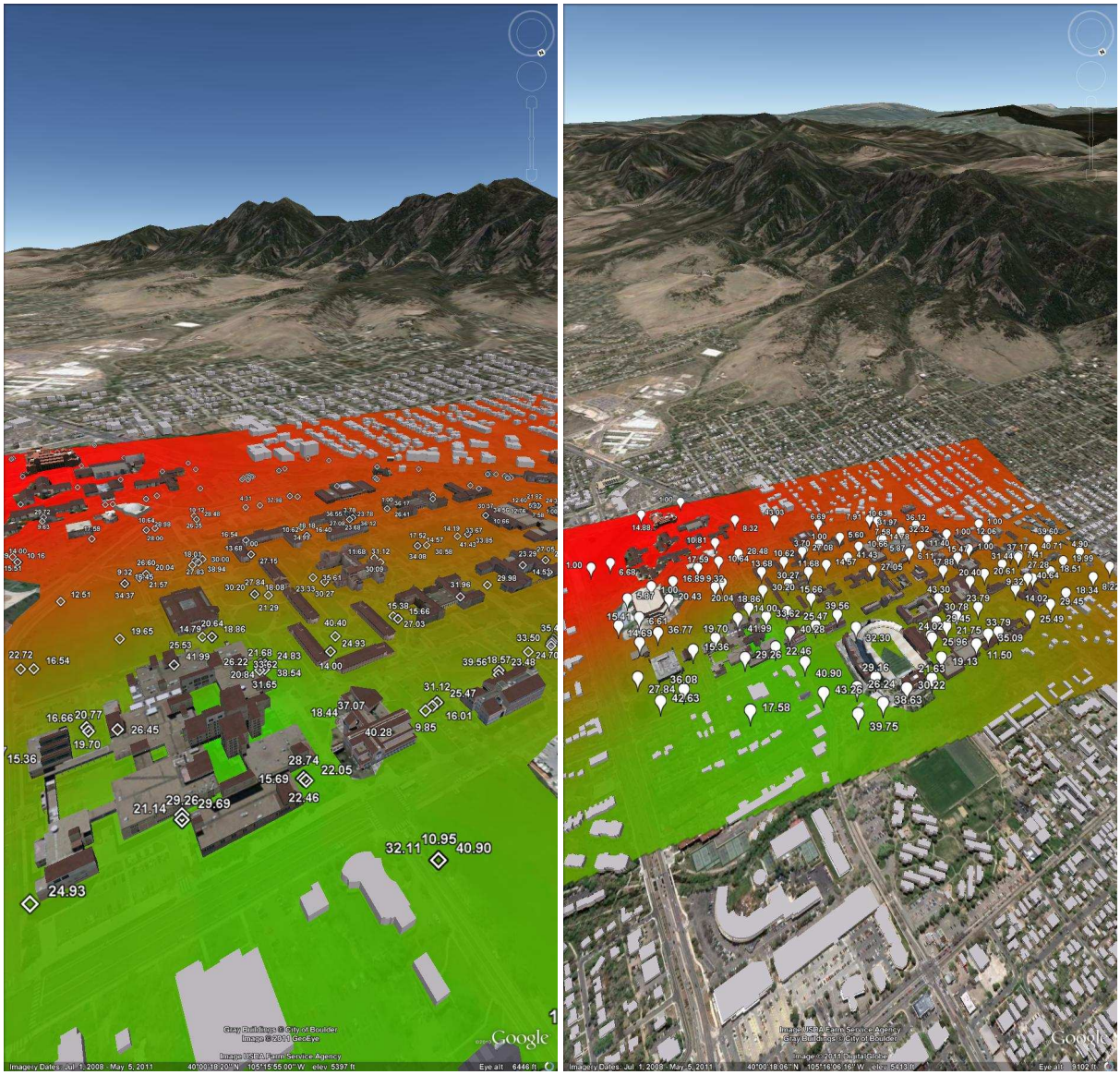
5.5 Summary and Conclusion

This chapter described the approach to geostatistical coverage mapping taken in this thesis, and practical adaptations made for the purpose of modeling the vagaries of RF propagation. Table 5.2 summarizes the best practices for geostatistical mapping derived in this thesis, and section C.5 provides source code implementing the core parts of the fitting and mapping functions. When performing variogram fitting, it was found that de-trending is necessary and can be accomplished by subtracting off predictions from a log/log fitted model. Null measurements can be included or not, using a constant low value where measurements could not be made. Variogram truncation is essential for a reasonable fit. The truncation value appears to be environment dependent, but largely consistent for APs in the same environment. Gaussian and cubic models perform well, with a nugget variance between zero and 40 wavelengths, and an imaging resolution of approximately 1 pixel per 5 meters (or 1 pixel per 20 when generating draft-quality maps). The approach described here utilized OK for Kriging and WLS for model fitting. However, Universal Kriging (UK) models, possibly utilizing anisotropic extensions are an interesting topic for future investigation. The next chapter will put the method proposed here to work in order to predict the coverage of production WiMax and LTE networks, as well as analyze the quantitative and qualitative efficacy of this approach to coverage mapping in the real world.



(a) (b)

Figure 5.13: Examples of coverage map (for CU WiMax cuEN node) overlayed on Google Earth orthoimagery, digital terrain, and 3D models.



(a) (b)

Figure 5.14: Examples of coverage map (for CU WiMax cuEN node) overlayed on Google Earth orthoimagery, digital terrain, and 3D models with measurement locations and values.

Initial Sampling Design	100m triangular lattice with random clustered samples within 40 wavelengths
Second Phase Sampling	None. See chapter 8.
Unreachable Points	Take measurement at nearest accessible location avoiding systematic bias
Repeated Measurements	1-3 per (regular or cluster) location to model small scale temporal variation in isolation
Negative/Non Measurements	Include with constant “noise floor” value
Detrending	Subtract off Log/Log fit
Variogram Fitting	Weighted Least Squares (although MLE or Method Of Moments (MOM) are acceptable as well)
Variogram Model	Gaussian or cubic
Variogram Truncation	Yes, environment/technology dependent
Kriging	Ordinary Kriging (Universal Kriging is a topic for *Future Work*)
Anisotropic Modeling	None. A topic for *Future Work*.
Nugget Tolerance	0 (or up to 40 wavelengths to smooth out fast fading effects)
Prediction Resolution	0.05 (fast) - 0.2 (slow) pixels per meter

Table 5.2: Summary of derived best practices for geostatistical mapping of wireless network coverage.

Chapter 6

Case Studies

This chapter continues the discussion that began in chapter 5 where an approach to geostatisitcal coverage mapping for wireless networks was described. In this chapter, those methods will be applied in two real-world applications: coverage mapping of WiMax and LTE networks on the CU campus. In addition, sections 6.2 and 6.4 will discuss two important additional topics: map combining, and modeling deviations from stationarity.

6.1 Case Study: University of Colorado WiMax

This section describes the first case study conducted specifically for the purpose of evaluating the efficacy of Kriging-based coverage mapping. This is an important step, since the exploratory examples described in chapter 5 made use of drive-test data, which may be ill-suited for coverage mapping due to a sampling bias towards "street canyons". The aim here is to map the coverage of five WiMax BSs deployed on the University of Colorado campus operating at 2.5 GHz within an educational spectrum license held by the University. An initial sample is taken on a uniform equalateral triangular lattice with a lag of 100 m. To constrain the data collection, measurements are confined to the main University of Colorado campus. Figure 6.1 shows the main campus along with points where samples were collected. The shape of the University is vaguely triangular, with the hypotenuse measuring 1.5 km and the shorter side measuring 1.1 km, giving a total measurement area of slightly more than $825m^2$.

Of the five WiMax BSs being studied, four are managed by the University of Colorado Office of Information and Technology (OIT) and primarily provide backhaul coverage to buses in and around Boulder

Figure 6.1: Map of University of Colorado and 100m uniform equalateral triangular sample. Measurements are limited to the main campus, which is outlined in red.

Name	Dir.	Freq.	Longitude	Latitude	Easting	Northing	AGL (m)
Gamow West (cuGW)	235	2530	-105.267778	40.008056	2017383.55	4582293.93	46
Gamow East (cuGE)	90	2520	-105.267778	40.008056	2017383.55	4582293.93	46
Eng East (cuEE)	120	2530	-105.263056	40.007222	2017808.74	4582284.17	34
Eng West (cuEW)	240	2510	-105.263333	40.007222	2017784.92	4582279.27	34
Eng North (cuEN)	0	2578	-105.263333	40.007222	2017784.92	4582279.27	34

Table 6.1: Specifications of five University of Colorado WiMax BSs.

[39]. The fifth is a Global Environment for Networking Innovation (GENI) testbed node used for research purposes [182]. Table 6.1 provides details about the location and configuration of each BS[1] . All nodes use a channel bandwidth of 10 MHz, have 90-degree sector antenna (excepting the GENI node which has a 120 degree sector), and operate at a nominal transmit power of 40 dBm. Two BSs are deployed on the Gamow Physics Tower (pointing east and west) and three on the Engineering Center tower (pointing north, east, and west). The cuGW and cuEE nodes are deployed on the same frequency, while the other BSs each have their own frequency. Because the spectrum analyzer used for measurement has no way of differentiating between different BSs on the same frequency and will simply record the strongest measurement observed, these two BSs will be measured and modeled as if they were a single BS with two antennas.

6.1.1 Measurement Apparatus and Procedure

In order to make measurements in arbitrary locations, which might not be accessible with a large vehicle, a measurement apparatus was constructed especially for this data collection campaign, built into a small cart. The cart provides a stable platform on two wheels and can be connected to a bicycle or used as a hand-cart. To collect measurements, an Anritsu MS2721B portable spectrum analyzer is used. This analyzer is unique in that it is both battery-powered and portable, as well as having the ability to demodulate WiMax transmissions. A netbook laptop running Ubuntu Linux is connected to the spectrum analyzer with a single Category 5 (CAT5) crossover cable. This laptop controls the spectrum analyzer using a series of Virtual Instrument Software Architecture (VISA) commands, which creates the ability for measurement scripting on the laptop. Two GPS devices are used to record position, one connected to the spectrum analyzer and

[1] Unless otherwise specified, all latitude and longitude coordinates are given in WGS84/EPSG:4326 and UTM coordinates in EPSG:32160.

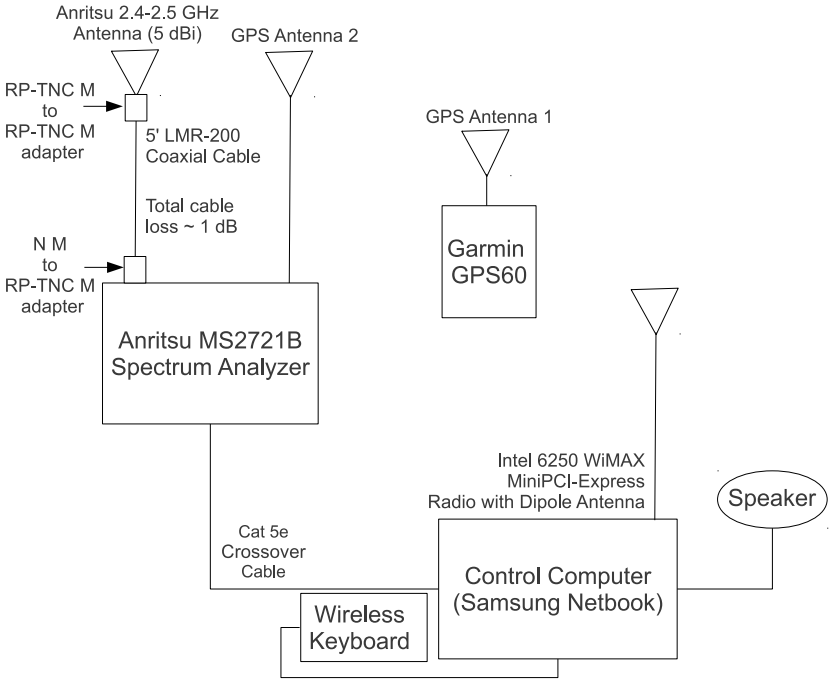


Figure 6.2: Diagram showing connectivity and specification of WiMax measurement cart devices.

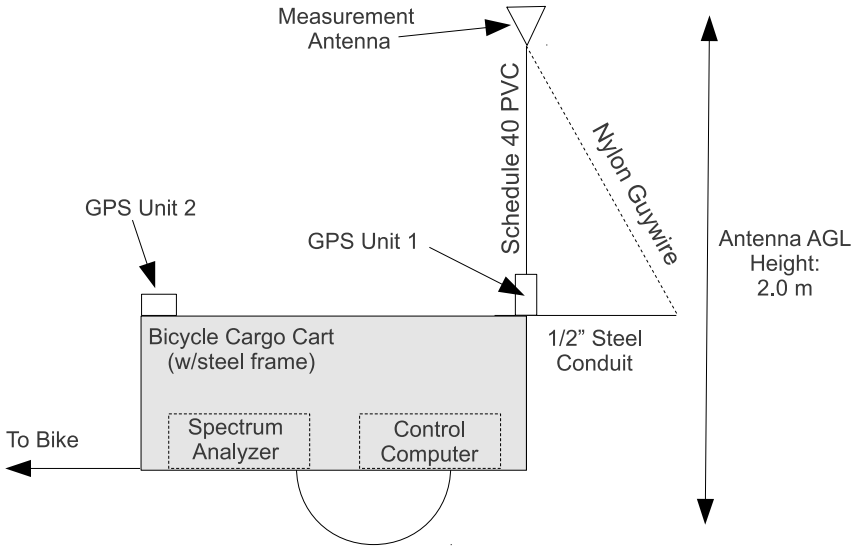


Figure 6.3: Diagram of WiMax measurement cart.

one a hand-held Garmin GPS60 device[2] . The measurement antenna for the spectrum analyzer is raised 2 m from the ground using a piece of schedule-40 Polyvinyl Chloride (PVC) (non-conductive) pipe, and attached with plastic cable ties. Although the cart itself is conducting, care is made to ensure that no metallic objects are in close proximity to the elevated measurement antenna. Figure 6.3 and 6.2 show the design and layout of the measurement cart.

The measurement effort focuses on four important first-order metrics of channel performance: Carrier to Interference and Noise Ratio (CINR), Relative Constellation Error (RCE), Error Vector Magnitude (EVM), and subcarrier spectrum flatness. CINR provides a measurement of pure received power above noise, calcuated from a clean carrier wave transmitted in the preamble of the WiMax frames. RCE and EVM quantify the amount of error in a binary or quaternary constellation plot, which provides a tight estimate of physical-layer error. Finally, subcarrier spectrum flatness is the amount of gain or attenuation on each of 52 (or more) subcarriers within the bandwidth relative to the mean signal strength. Using the spectrum flatness data, it is possible to calculate Effective Signal to Noise Ratio (ESNR), the metric shown in [90] to be a strong predictor of actual network performance (as compared to the more traditional metrics such as SNR and RSS).

Before begining measurement, a policy must be defined for locating and measuring at sample sites. After some experimentation with direct location using a GPS device, a simple solution was chosen involving a printed map similar to 6.1. Each site is visited without any particular order. In the event that it is impossible to make a measurement at the site, either because it falls in an inaccessible (e.g., fenced) area or within a building, a measurement is made at the closest point (by straight line distance) that is measureable. Although there is some random error associated with locating points (due to GPS accuracy, point finding, and obstacles), this error is not harmfully aligned with any environmental feature and instead amounts to random jitter about the uniformly selected sample sites (which some spatial sampling studies have actually purposely advocated).

At each measurement location, a wireless keyboard is used to manage the control computer (which keeps the experimenter away from the apparatus, preventing them from interfering with the measurements

[2] A hand-held GPS device was chosen after finding the Anritsu's GPS reception to be weak and unreliable.

themselves) and the control computer provides feedback through an amplified speaker utilizing text-to-speech synthesis software. At each point, three repeated measurements were made of downstream system performance using various metrics. At a subset of points, additional clustered measurements were taken within a 40 wavelength radius of each true point. The combination of repeating measurements in time and space allows for accurate estimation and averaging of intrinsic channel variability due to small scale fading effects. The device first picks a given channel (carrier frequency) and then records all metrics for each measurement in turn. Then it switches to a different channel and repeats. While the device is performing measurements, the instrumenter uses the handheld GPS device to record the current position, sample location (each sample site is assigned a unique identifier), and GPS accuracy. At the end of a measurement effort (typically when the analyzer's battery is flat), the cart is returned to the lab for charging and data offload. The spectrum analyzer stores measurements in a proprietary, but plaintext, format that can be easily parsed.

6.1.2 Comparison of Performance Metrics

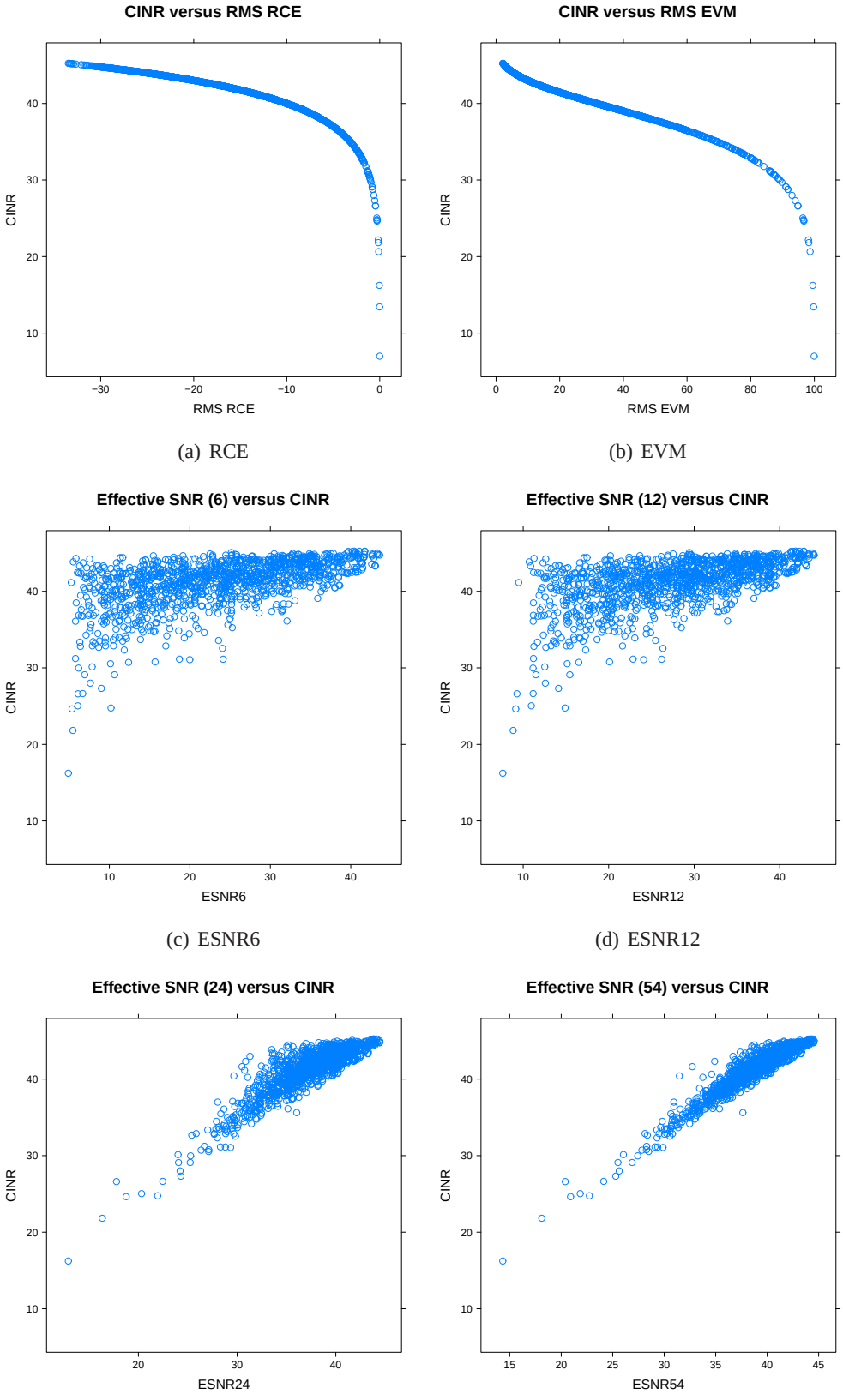
In this measurement campaign, several performance metrics besides the classic signal strength or SNR-equivalent metrics were collected. One question that naturally arises is: are these more robust metrics trivially correlated with simple and easy to collect metrics such as CINR? Figure 6.4 plots the relationship between CINR and each of the other metrics studied. RCE and EVM appear to be a simple (but nonlinear) function of CINR, at least as calculated by the spectrum analyzer used. There are several ways that EVM can be calculated from the constellation plot and observed power of constellation points, and it appears that the Anritsu spectrum analyzer is calculating EVM from CINR or vice versa. RCE is calculated directly from the EVM value and hence is equivalent. Given this, RCE and EVM do not provide novel information above and beyond what is provided by the CINR measurement. It is worth noting that in the process of data collection, a complete constellation plot is recorded for each measurement so these metrics could be calcuated *ex post facto*. The relationship between ESNR and CINR is less trivial, especially for the lower (Phase Shift Keying (PSK) modulation based) bitrates. The higher bitrates, which use Quadrature Amplitude Modulation (QAM), tend to have a fairly well-defined linear correlation with CINR. This suggests that in cases where information about spectrum flatness is unavailable, ESNR54 can be roughly approximated using

CINR measurements.

Knowing which metrics provide unique information about the channel, a natural followup question is: are these metrics correlated with application-layer performance? To answer this question, a measurement campaign was devised to perform throughput tests to the the cuEN BS, the only one of the four BSs which is not in production use (and hence could produce clean throughput measurements), and was accessible at the time of measurement. In order to use a sufficiently diverse data set, measurements from a random sample of points around campus (which typically are NLOS to the antenna) were combined with measurements at regularly spaced intervals down a street in the LOS path of the BS antenna. At each point, a three-sample measurement was taken using the method described in the previous section. At the same time a netbook running Microsoft Windows XP, using a Accton Wireless Broadband Corporation (AWB) US210 wireless network adapter[3] was used to collect a series of throughput measurements. Because it is difficult to fully saturate a high bandwidth link using an application-space packet generator (in Windows, especially), the choice was made to rapidly generate a large number of User Datagram Protocol (UDP) packets at a selection of packet sizes. The open-source tool nping [204], was used to generate 5,000 packets packets each using a payload of 80 bytes, 256 bytes, 512 bytes, 1024 bytes, and 1400 bytes. This results in 25,000 packets being transferred at the maximum rate possible, which amounts to 17,460,000 bytes, or 16.64 megabytes. To ensure that only the wireless system was being tested, the endpoint used was the BS Access Service Network (ASN) gateway controller, which was running Ubuntu Linux. Both the netbook and the ASN system collected a complete trace of sent and received packets using the tcpdump tool (or Windump in the case of Windows) [219, 221]. At the start of a throughput test, the measurer would attempt to make a connection to the BS, if a connection was obtained, the test would start by sending the upstream packets. Upon receipt of the last upstream packet, the daemon on the ASN server would wait five seconds and then begin sending an identical volly of packets downstream. The tracefiles were analyzed *ex post facto* to determine the loss rates and per-payload-size throughput.

To analyze how well physical-layer metrics collected with the spectrum analyzer predict application layer performance, an ANOVA was performed against each metric and upstream and downstream through-

[3] This adapter was tested and approved by the GENI WiMax project.

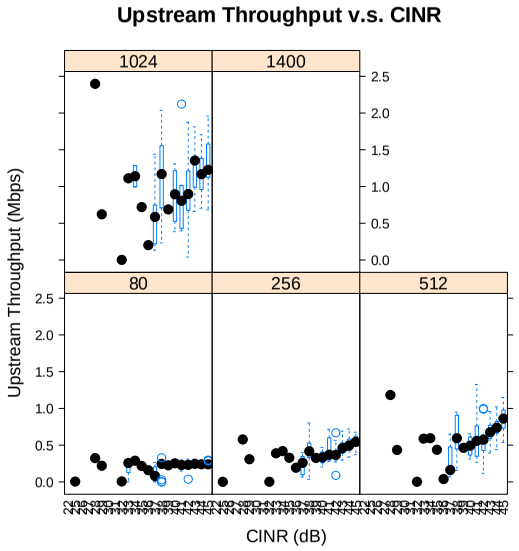


(a) RCE (b) EVM

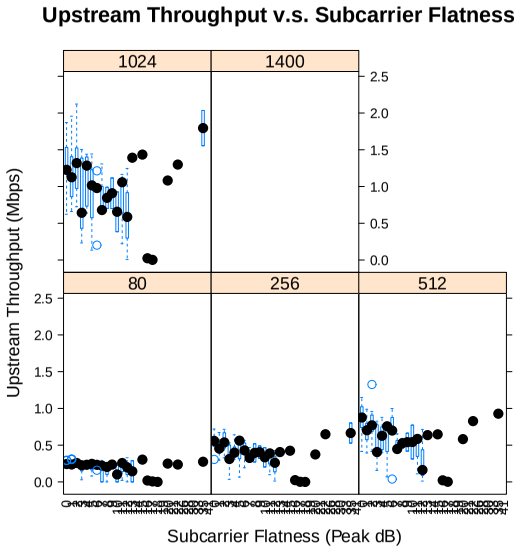
(c) ESNR6 (d) ESNR12

(e) ESNR24 (f) ESNR54

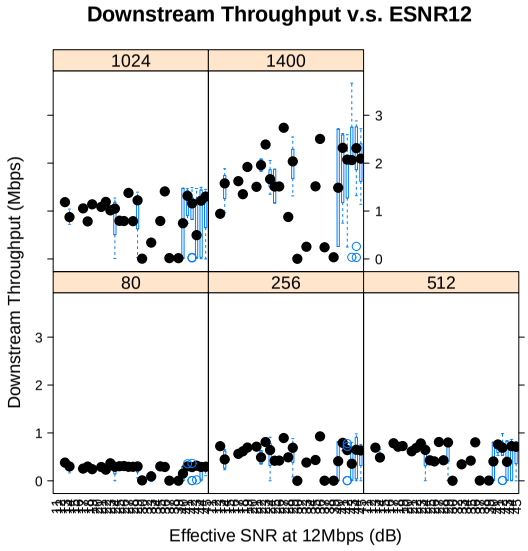
Figure 6.4: Correlation between various metrics.



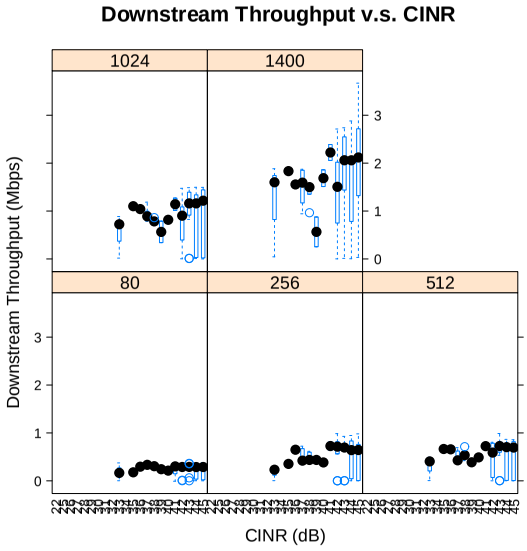
(a) D-Tput v.s. CINR



(b) U-Tput v.s. SCF



(c) D-Tput v.s. ESNR



(d) D-Tput v.s. CINR

Figure 6.5: Correlation between WiMax throughput and metrics that are meaningfully correlated with application-layer performance.

put. For upstream throughput, the most significant correlations were present with CINR and Subcarrier Flatness (SCF), with a $p - value << 0.05$ for each and a $F - value$ of 28.10 and 12.45, respectively. Downstream measurements are more elusive—the only metric that has a significant correaltion with a $p - value < 0.05$ was ESNR, with an F-value of 4.86. Figure 6.5 shows the relationship of these metrics. Clearly there is a linear correlation in each case, however the correlations are very noisy. This noise is likely a result of unreliable components, in addition to the expected environmental noise. For instance, the ASN gateway server had a tendency to become loaded when generating or receiving a volly of packets and drop some frames. Similarly, the Winpcap library appears to be much slower than the linux-based equivalent and simply cannot generate packets at the same rate. Some packets, such as the smallest ones, were silently dropped by the AWB usb dongle (whose queue presumably overflowed when trying to send the 80 byte packets, and whose Maximum Transmission Unit (MTU) size appears to prohibit the 1400 byte packets). Despite this unavaoidable noise, there is a clear and distinct correlation present, which supports the use of CINR (which differs from SNR in that it is calculated from a clean carrier signal sent during the packet preamble) and spectrum-flatness-based metrics such as SCF and ESNR, supporting the findings of Halperin *et al.* in [90]. These results also suggest that, in terms of CINR, there is a clear division in performance above and below 40 dB, a threshold which can be used much in the same way Robinson et. al use 20 dB for WiFi networks when locating coverage “holes” [200].

As a final curiosity, GPS receiver accuracy is compared to CINR in figure 6.6. While making measurements, locations with the highest signal seemed to also have a high GPS precision (small accuracy value). This stands to reason since GPS also operates at 2.5 GHz and is effected by multipath interference from obstructions, in particular buildings and street canyons in urban environments. However, as shown by the figure, in practice the correlation is weak and noisy (certainly too noisy for planning purposes).

6.1.3 Possible Sources of Systematic Sampling Error

During the measurement campaign, three individuals used the cart to make measurements. Although all three measurers were collecting measurements using the same procedure, one possible source of systematic error is from the measurers themselves. Figure 6.7 shows the location error and GPS accuracy as

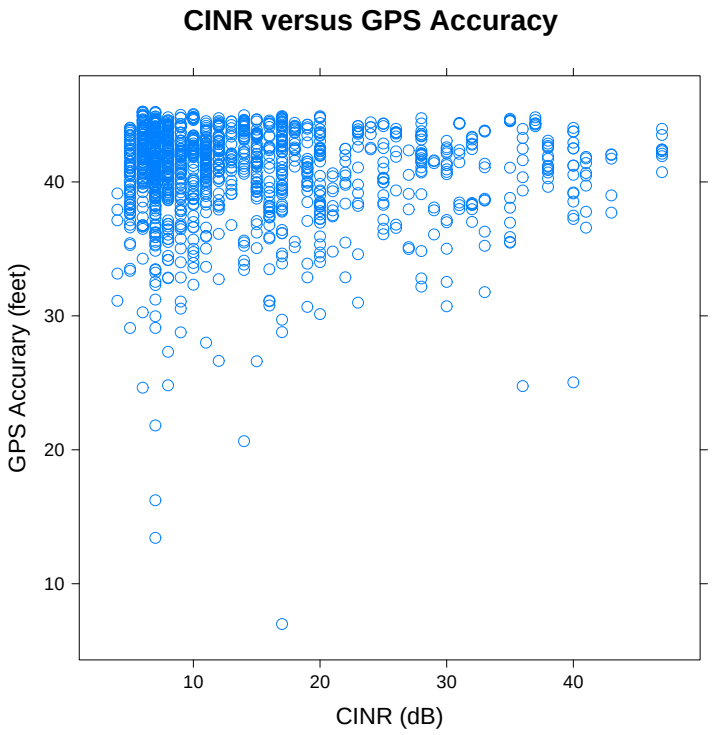


Figure 6.6: Correlation between CINR and GPS accuracy

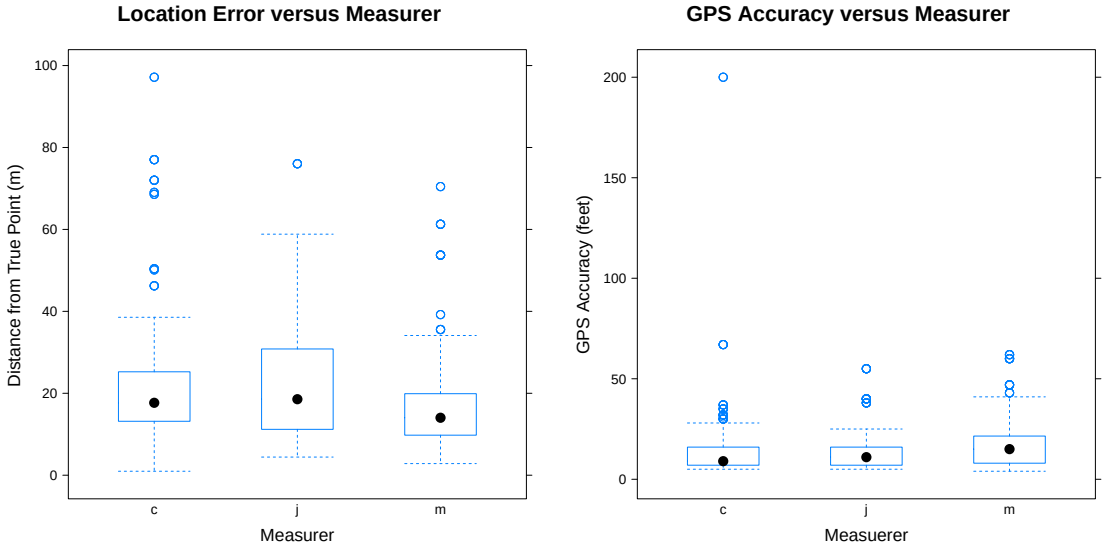
a function of which measurer performed the measurements. There appears to be no discernable correlation and hence we do not need to correct for this bias in subsequent analysis. It is worth noting that some measurements are distant from their intended location. As discussed above, this occurs when a point is unreachable in practice. So long as the new measurement point is as close to the original measurement location as possible and there is no systematic error or systematic terrain alignment, these deviations should not effect the quality of the sample.

6.1.4 Spatial Data Characterization and Variogram Fitting

Figure 6.8 shows measurements taken for the cuEN BS, highlighting four metrics of interest: path loss computed from CINR, ESNR6 (ESNR for 6 Megabits per second (Mbps)), and ESNR54 (ESNR for 54 Mbps). The other four BSs produce similar plots. All four metrics produce a similar spatial distribution of values with large path loss or error values to the southwest and smaller (better) values to the north. All metrics have different value distributions, but the ESNR54 and and CINR metrics appear to share the same basic skewed lognormal shape. Figure 6.8 shows the fitted relationship between path loss and distance for the three SNR-like metrics. The fits are not fantastic, but appear to at least account for some basic trend, which we can be removed to improve the efficacy of the Kriging process. The group of measurements around 155 dB are the “inferred” null measurements. For the SNR-like metrics, 1.0 is used for the null measurement and for EVM, 100 (i.e., 100% probability of error) is used.

After de-trending and accounting for “null” measurements, the next step is variogram fitting. Figure 6.10 shows the fitted variograms for cuEN using the same procedure as descibed in the previous section. As before, the fits are truncated. This time fits are truncated at 1.4 km, since this is approximately the width of the campus, and measurements further apart than that are unlikely (or erroneous). Because nugget variance was modeled explicitly with clustered measurements, the nugget tolerance is set to 0. Table 6.2 and provides the fitted variogram parameters. In this data set, the best fits are generally truncated, but without null samples at locations where a measurement was not observed. The best fits are split fairly evenly between Gaussian and cubic models.

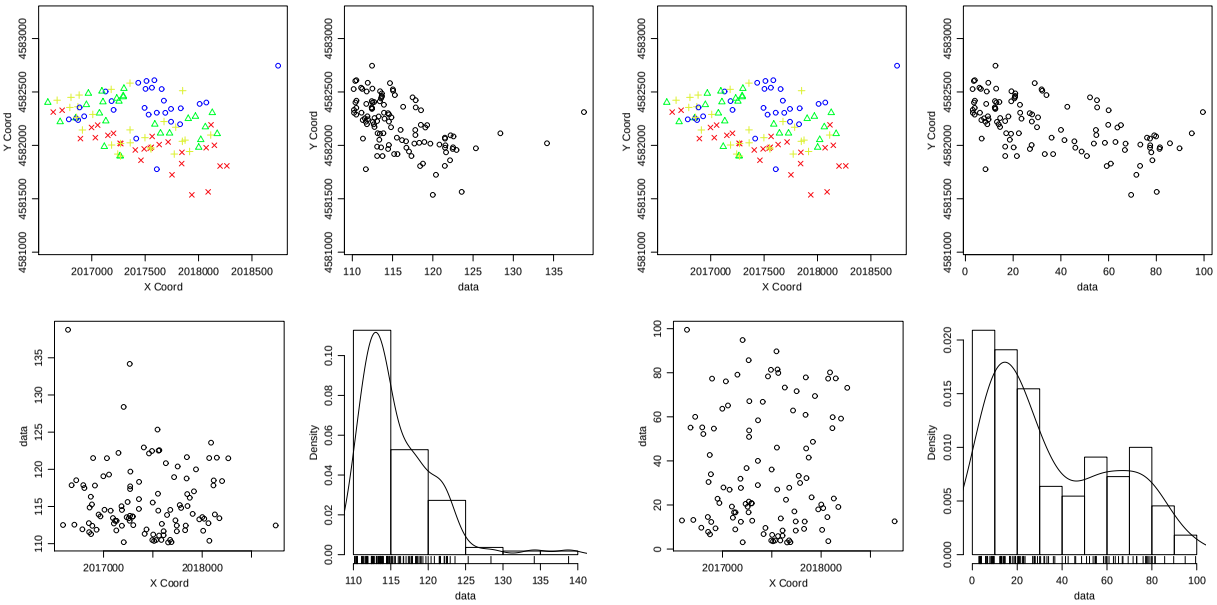
As described in section 5.4.5, to determine the goodness of these fits a 10-fold cross validation is done



(a) Location Error

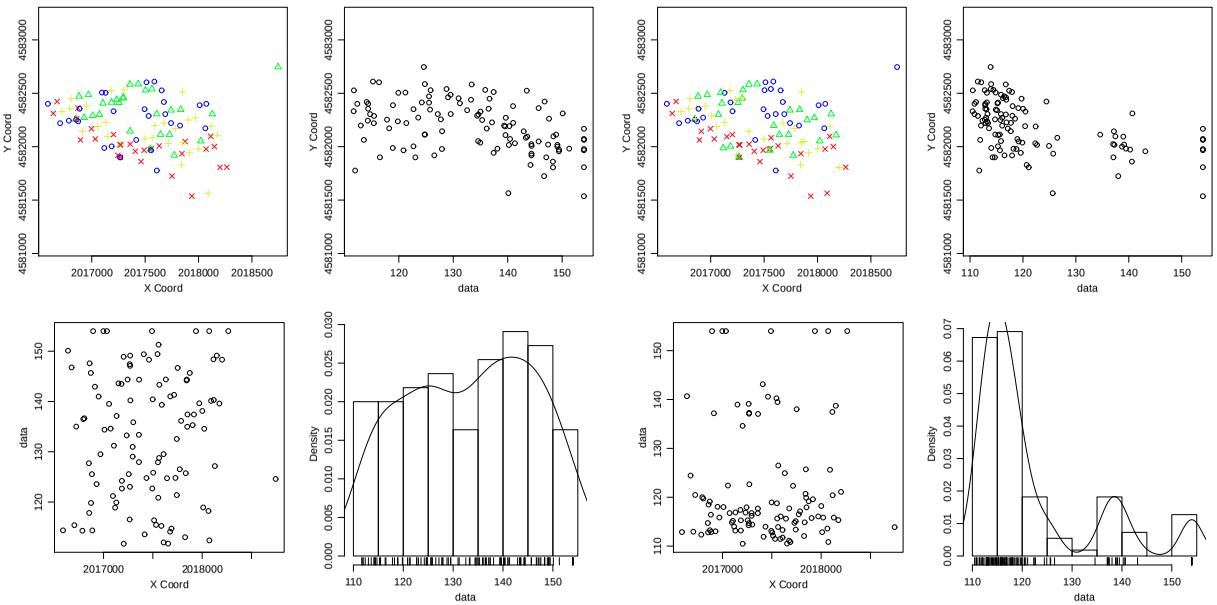
(b) GPS Accuracy

Figure 6.7: Measurement location error and GPS accuracy as a function of the measurerer.



(a) PL from CINR

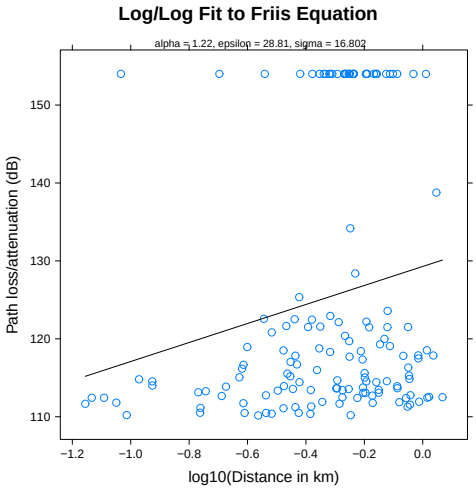
(b) EVM



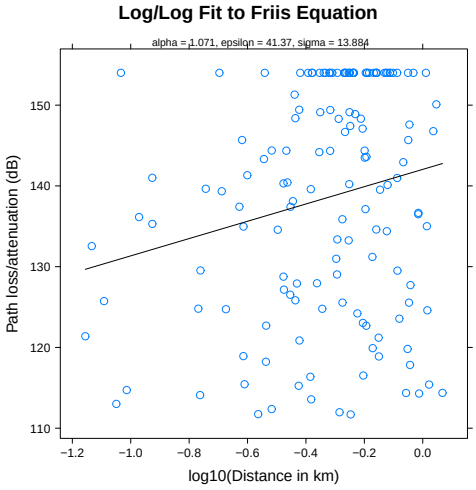
(c) PL from ESNR6

(d) PL from ESNR54

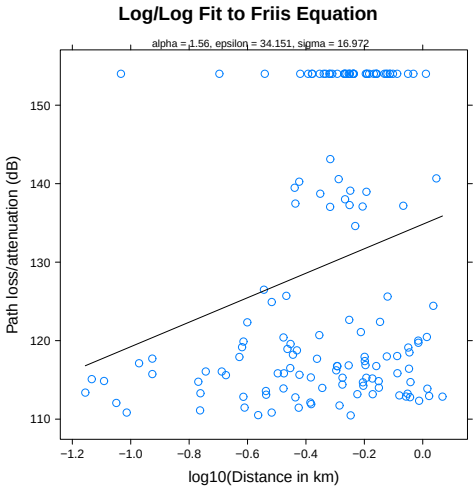
Figure 6.8: Measurements for CU WiMax cuEN (GENI) node.



(a) PL from CINR

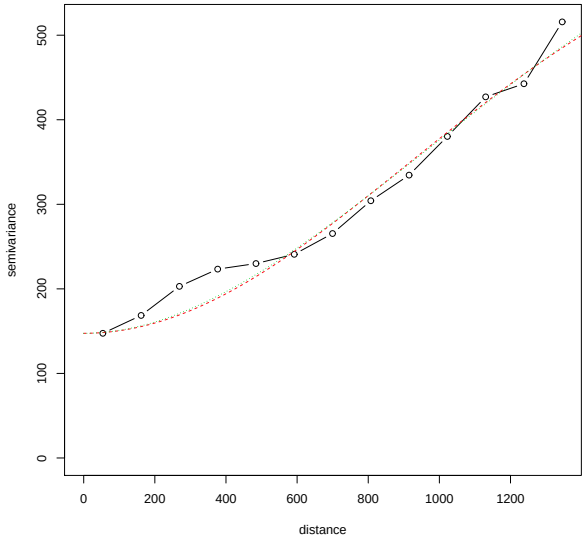


(b) PL from ESNR6

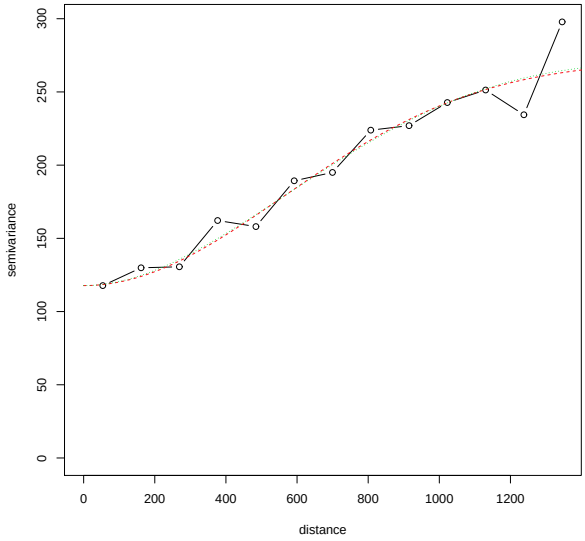


(c) PL from ESNR54

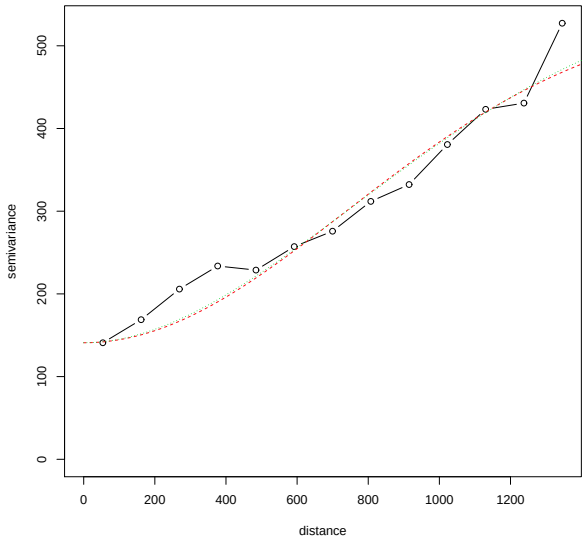
Figure 6.9: De-trending fits for the CU WiMax cuEN (GENI) node. Only the metrics that can be converted to path loss and de-trended (i.e., SNR and equivalents) are shown.



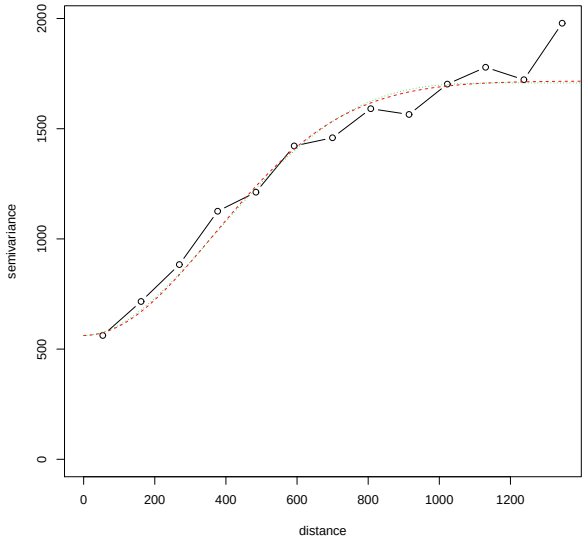
(a) CINR



(b) ESNR6



(c) ESNR54



(d) EVM

Figure 6.10: Empirical variogram and fits of four metrics for CU WiMax cuEN (GENI) node.

by random-exclusion of 10% of points, or a maximum of 50. This results in two metrics of predictive performance: RMSE (error of the predictions) and square-root of Kriging variance (residual error of the model). These metrics are averaged across the 10 folds. Overall, the fits for the CINR are quite good, acheiving a RMSE of 2.03 for the cuGE BS, 4.09 for cuEN, and approximately 2.8 for the other two BS. This error is excellent, even by the standard of typical repeated-measures variance for outdoor urban environments [198]. The predictions for the ESNR metrics are less strong, ranging from 5 to 11 dB, which is still well under the typical performance of the *a priori* models described in chapter 3. Why ESNR fits less cleanly is not clear, although one explanation may be that the additional degrees of freedom it considers causes it to involve more intrinsic variability. The EVM metric, which was not de-trended, fits fairly poorly. Given that EVM can be computed from CINR directly, a better approach to mapping EVM is probably to map CINR and then translate the resulting map. The mean Kriging variance, which describes residual error in the model instead of predictive accuracy, tends to mirror the other metric. A final metric of improvement is given in the right-most column: the gain (reduction in error) as compared to the residual error of an explicit log/log fit to the measurements. The geostatistical fitting method produces a reduction in residual error for all metrics and all BSs.

6.1.5 Mapping with Ordinary Kriging

Figure 6.11 shows the resulting map for the cuEN node, which is a 120-degree sector propagating to the North. As might be expected, it appears to cover the 120-degree region to the North quite well and less well behind. While there is a clear difference between the predictions of the various metrics, the ESNR metrics are clearly predicting a less uniform propagation. The excess maps (before the de-trended values were added back), may actually provide the most useful information, since they show the deviation from a standard power-law relationship. There appears to be a very large positive excess directly behind the transmitter. This makes sense, as there is a large concrete wall preventing line of sight in that direction. As will be discussed in chapter 8, this is an area of particular interest. Interestingly, the ESNR maps push this high-point of excess further southeast, towards an area of campus shadowed by a hill, in the back-lobe of the antenna. The variance maps demonstrate the region of confidence for the map (the center red region)

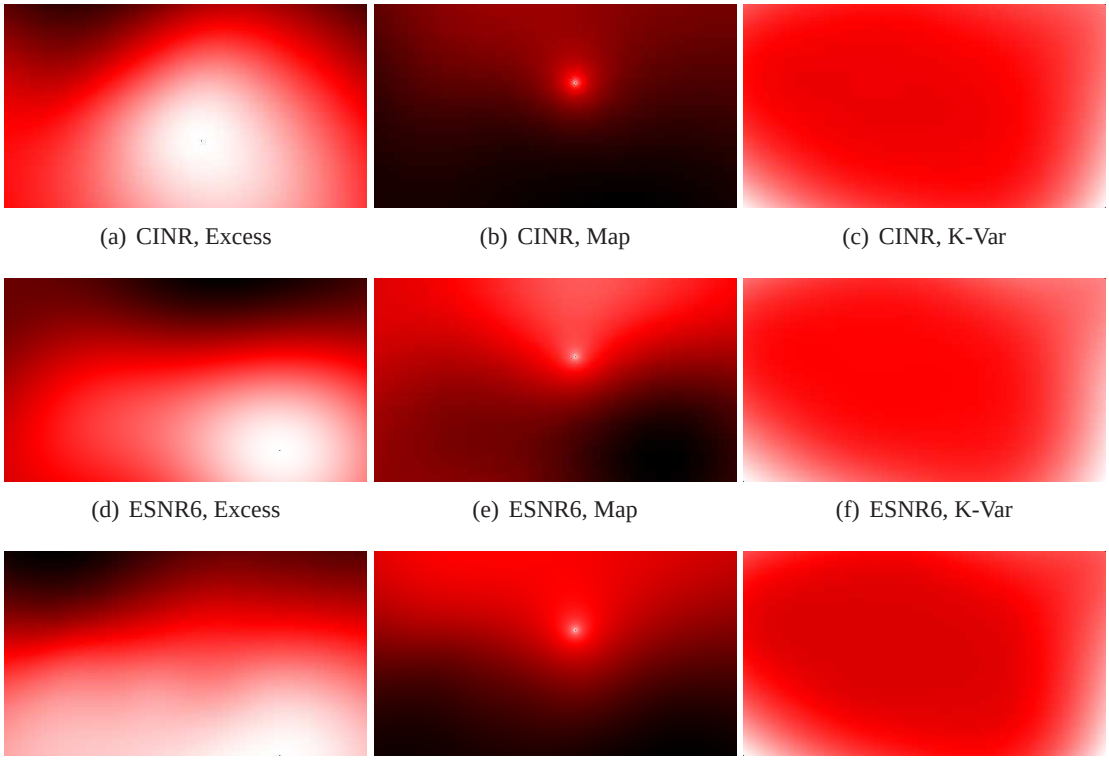
BS	Metric	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
cuEW	100wimaxEVM	gaussian	697.13	199.12	351.34	150	FALSE/FALSE	15.00	16.05	N/A
cuEW	100wimax	cubic	1839.69	3.99	19.38	150	FALSE/FALSE	2.16	2.75	17.54
cuEE/cuGW	100wimaxESNR54	cubic	2183.29	115.19	81.75	147	TRUE/FALSE	11.09	9.27	5.88
cuEE/cuGW	100wimaxESNR6	cubic	1253.62	91.27	45.66	147	FALSE/FALSE	9.95	9.50	2.68
cuEE/cuGW	100wimaxEVM	cubic	771.36	259.17	396.46	147	TRUE/FALSE	17.68	15.91	N/A
cuEE/cuGW	100wimax	gaussian	541.94	8.48	9.30	147	TRUE/FALSE	3.04	2.87	12.65
cuGE	ESNR54	gaussian	2340.33	34.11	437.08	182	TRUE/FALSE	5.93	6.91	7.15
cuGE	ESNR6	gaussian	380.27	49.67	39.18	182	TRUE/FALSE	7.34	7.48	2.99
cuGE	EVM	gaussian	310.67	138.75	321.18	182	TRUE/FALSE	12.67	12.25	N/A
cuGE	CINR	cubic	1711.76	6.39	12.31	182	FALSE/FALSE	2.61	2.03	9.80
cuEN	ESNR54	cubic	1530.11	72.81	108.83	146	TRUE/FALSE	9.00	9.83	7.97
cuEN	ESNR6	gaussian	746.71	118.74	76.04	146	TRUE/FALSE	11.22	11.21	2.67
cuEN	EVM	cubic	751.21	444.98	357.14	146	FALSE/FALSE	22.84	21.11	N/A
cuEN	CINR	cubic	1304.05	14.22	20.04	146	TRUE/FALSE	4.00	4.09	12.80

Table 6.2: Best fit statistics for variogram fitting of CU WiMax BSs.

with confidence fading towards the borders where less information is available.

As another example, figure 6.12 shows the final maps for the cuGE node. This BS is a 90-degree sector pointing east, and as a result the propagation seems to favor that direction; however, there are clear and significant shadows to the west. This stands to reason, as this transmitter is high on a tower, so the region due west is not only in the back-lobe of the antenna, but may also be in the region under the half-power point (i.e., under the line of sight level with down-tilt) even for the back-lobe. Figure 6.13 shows a simplified view of the coverage of these BSs, where a threshold has been set at $CINR = 40$, based on the measurements described in section 6.1.2. From this simplified picture, it is clear to see the region of campus covered by each BS. This picture also highlights the relative complexity of the contours in the underlying maps, which are sometimes hard to discern in a colormap, even using both hue and luminance.

As a final metric of performance for these maps, each map is compared to a random sample of measurements taken around campus to see how well the maps are able to predict points *in between* the sample grid. For this experiment, 100 random sample locations were chosen and tested sequentially. Measurements were only made of the cuEN node for this test. When comparing those measurements to the predictions for the cuEN node, using the CINR metric alone, there is a RMSE of 4.71, slightly higher than that found with cross validation, but still quite good overall. If we look to the ability to predict "holes", as was the goal of Robinson's work in [200], the map acheives an accuracy of 69%, involving five false positives, and eight false negatives, out of 42 total test points. Since this map was produced using 110 unique measurements in a 2.54 km^2 area, it required 43.3 measurements per square km. As one point of comparison, at the same sampling density, Robinson's iterative heuristic refinement proposal acheived approximately 80% accuracy. However, there are important semantic differences between the model presented here, which computes a smooth prediction of metric coverage over the entire region, and Robinson's proposal, which is designed to find threshold boundaries in streets, using drive-test measurements. Given this, acheiving a prediction accuracy only 10% less than Robinson's proposal, while maintaining all of the other benefits of the geostatisitcal method, seems like an encouraging result.

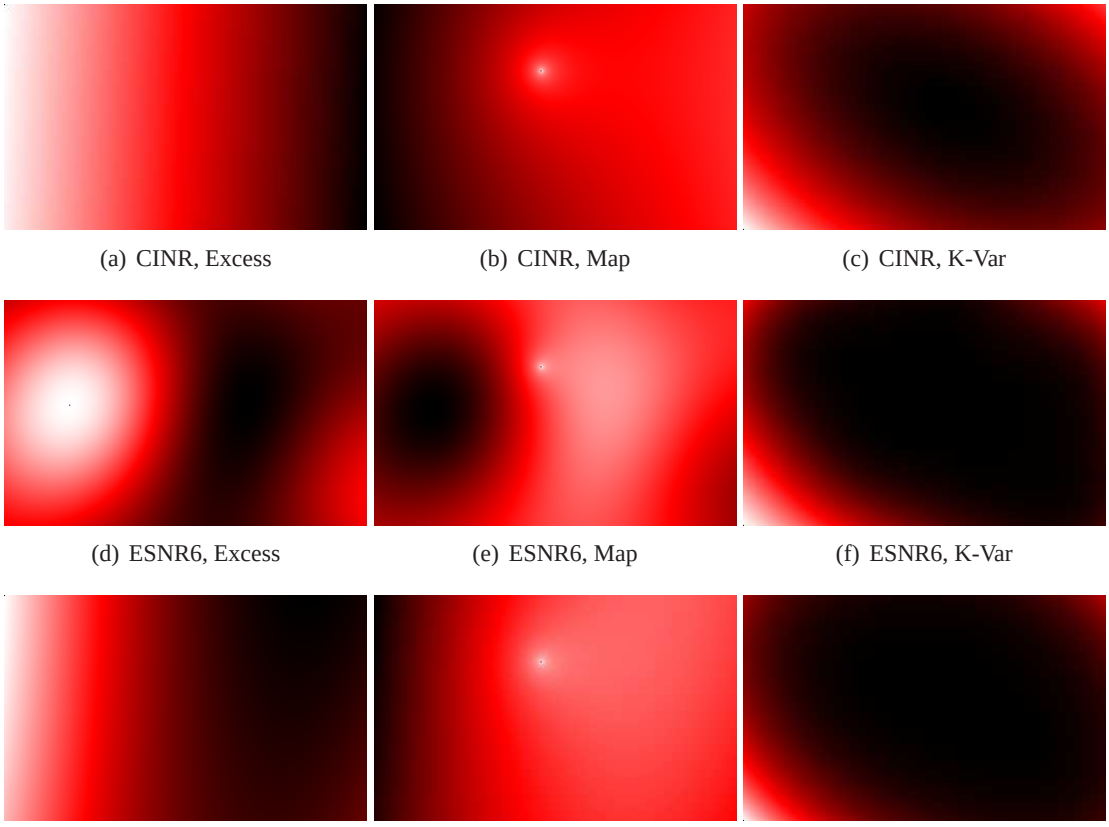


(a) CINR, Excess (b) CINR, Map (c) CINR, K-Var

(d) ESNR6, Excess (e) ESNR6, Map (f) ESNR6, K-Var

(g) ESNR54, Excess (h) ESNR54, Map (i) ESNR54, K-Var

Figure 6.11: Maps for cuEN node. The left maps show the excess (residual after trend is removed). The center maps show the re-trended signal map. The right maps show the residual kriging variance of the other maps.

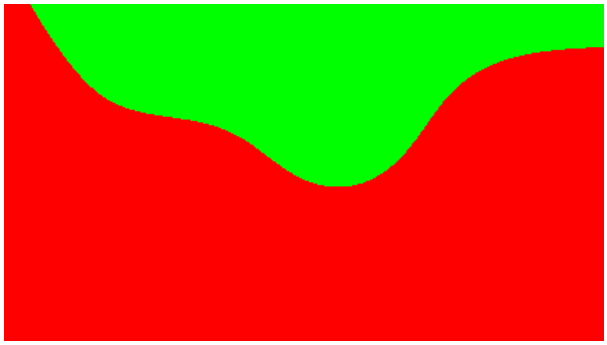


(a) CINR, Excess (b) CINR, Map (c) CINR, K-Var

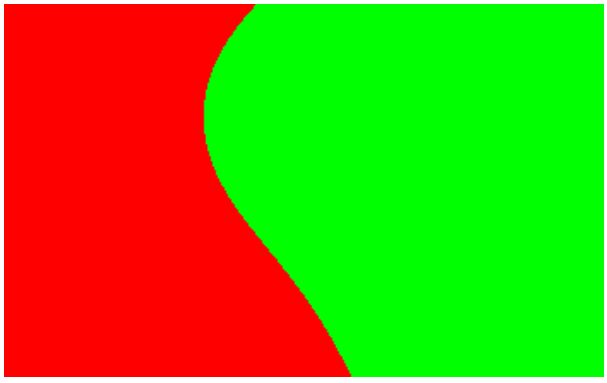
(d) ESNR6, Excess (e) ESNR6, Map (f) ESNR6, K-Var

(g) ESNR54, Excess (h) ESNR54, Map (i) ESNR54, K-Var

Figure 6.12: Maps for cuGE node. The left maps show the excess (residual after trend is removed). The center maps show the re-trended signal map. The right maps show the residual kriging variance of the other maps.



(a) cuEN



(b) cuGE

Figure 6.13: Binary coverage maps with the threshold of CINR=30

6.2 Map Combining

The previous section discussed how to generate per-BS maps of coverage using geostatistical techniques, guided by the example of the CU WiMax network. This section seeks to understand how measurements from multiple BSs can be combined to produce a composite map.

6.2.1 Data Combining

For a network that contains measurements from many BSs, a natural question is how a composite map can be created for the entire network. This can be accomplished either by fitting and Kriging the entire set of measurements or by fitting and Kriging measurements from each BS separately and then combining the resulting maps. The first approach is the most conceptually straightforward, but has some problems. Combining measurements from multiple APs may produce a map with a large amount of per-location variation, possibly with co-located points of drastically varying value. Exactly co-located measurements of differing value can produce unsolvable Kriging equations and must be "jittered" to create a solvable equation with a unique solution. In the end, this approach can result in a map that is difficult to interpret and has a large error margin. Consider figure 6.14 and 6.15, which show the fitted variograms and resulting maps for all the CU WiMax data combined, adding a jitter of up to 20 wavelengths to co-located measurements. The fitted variogram is relatively flat and has a large nugget variance at 250, implying an intrinsic variability at co-located points of more than 15 dB (the square root of the nugget variance). This is due to the fact that co-located points may be collected from different transmitters, and as such may have wildly differing values. However, this is not to say that the resulting map cannot be useful in spite of this basic error. Cross validation of this fit provides a mean RMSE of 3.72, a mean standard error (Kriging variance) of 3.79 and a gain of 17.64 dB over the residual error of a straight line fit through the measurements.

6.2.2 Map Combining

As an alternative to the data-combing method, Kriged maps can be combined *ex post facto*. Combining multiple Kriged maps that involve predictions in overlapping or partially-overlapping spatial regions

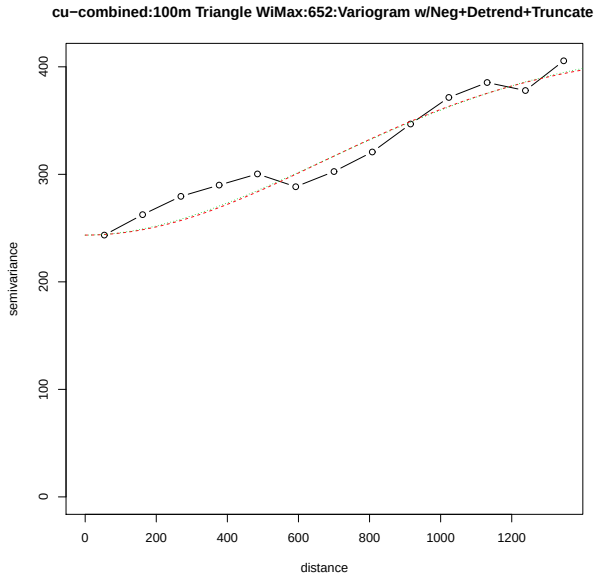


Figure 6.14: Empirical variogram and fits of four metrics for the combined CU WiMax measurements.

(a) CINR, Excess (b) CINR, Map (c) CINR, K-Var

Figure 6.15: Kriged maps for combined CU WiMax measurements using the CINR metric.

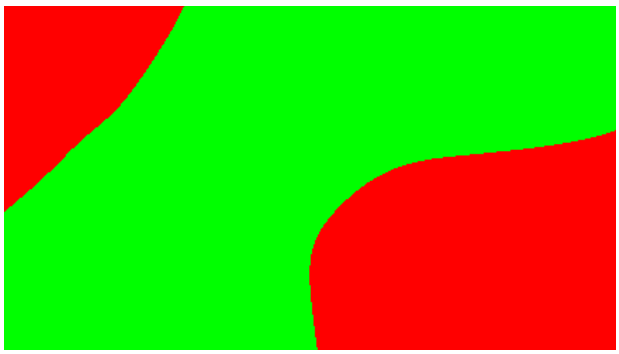


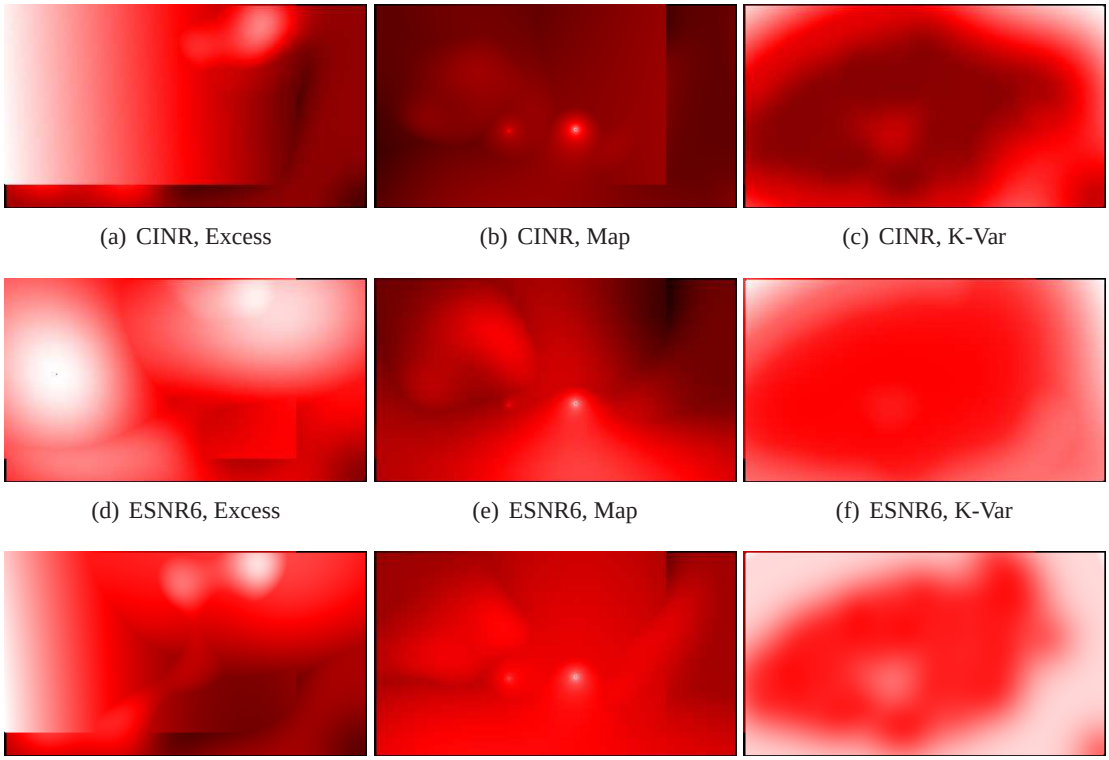
Figure 6.16: Binary coverage map for the combined CU WiMax measurements.

involves basic geospatial image tiling and combination. A basic two-pass method that first reads in all the map files to determine the total extent of the image, and then overlays the images, combining values at pixels as necessary, is used here. Algorithm 3 outlines this algorithm, where the result of computation is a matrix named “final” and the function “combine” is responsible for converting a vector of up to N values (some of which may be NULL). There are many maps that can be combined this way, the most obvious is to take the maximum value for SNR-like metrics or the minimum value for EVM-like metrics. In threshold-based combining, the number of transmitters whose interpolated signal is above 40 dB CINR (or below 60% in the case of the EVM metrics) is mapped.

Figure 6.17 shows the map-combined maps for the CU WiMax measurements. As compared to the data-combined maps, in these maps the strong signal associated with each of the two BS sites is clearly displayed with the Engineering Center Office Tower (ECOT) (right) tower showing a stronger signal since it houses three BSs instead of two. These maps show a more complex landscape of combined path-loss than the data-combined maps and bring to light some interesting observations. Chiefly, there appears to be two predominant shadows present, one due west of the Gamow (left) tower and one to the northwest of both towers. Although the source of these two shadows is unclear, and indeed it may not be possible to know for certain, it seems likely that the shadow due west of the Gamow tower is due to a misconfiguration of the downtilt on the cuGW node, since it should be covering this direction. However, since it is placed on top of an eight-story tower, the downtilt may be insufficicient to cover the region directly west, below the antenna. Because this network is used to provide access primarily to buses, and one of the bus routes passes through this shadow on Colorado street, this may actually be a problem which deserves some attention. Figure 6.18 shows the threshold-based combining map where the gradient from red to green corresponds to the number of BSs providing at least a signal of 40 dB for CINR and 20 dB for ESNR. These maps differ substantially, and as a result are somewhat difficult to interpret out of context to the underlying environment. However, even at a high level these maps communicate areas lacking coverage (bright red), and areas of possible co- and cross-channel interference (bright green) where all four transmitters have strong signal.

Algorithm 3 Partial Overlay Map Tiling

$N \leftarrow$ number of maps we are combining
$minx \leftarrow$ minimum x coordinate of all maps
$miny \leftarrow$ minimum y coordinate of all maps
$maxx \leftarrow$ maximum x coordinate of all maps
$maxy \leftarrow$ maximum y coordinate of all maps
$res \leftarrow$ resolution of all maps in pixels per meter
$width \leftarrow \lceil (maxx - minx) * res \rceil$
$height \leftarrow \lceil (maxy - miny) * res \rceil$
$mat \leftarrow$ a NULL-filled matrix of dimensions width $\times$ height $\times$ N
$n \leftarrow 0$
for each map m **do**
    $top \leftarrow$ topmost y coordinate of this map
    $left \leftarrow$ leftmost x coordinate of this map
    $xshift \leftarrow \lceil |top - maxy| * res \rceil$
    $yshift \leftarrow \lceil |left - minx| * res \rceil$
    $row \leftarrow height - xshift$ ▷ Loop over rows from top of image to the bottom
    for each row r in m **do**
        $row \leftarrow row - 1$ ▷ Loop over columns from left to right
        $col \leftarrow xshift$
        for each column c in r **do**
            $col \leftarrow col + 1$
            $mat[col, row, n] \leftarrow$ value at $m[c, r]$
        end for
    end for
    $n \leftarrow n + 1$
end for
$final \leftarrow$ a zero-filled matrix of dimensions width $\times$ height
for 0 to $width$ as x **do**
    for 0 to $height$ as y **do**
        $final[x, y] \leftarrow combine(mat[x, y])$
    end for
end for

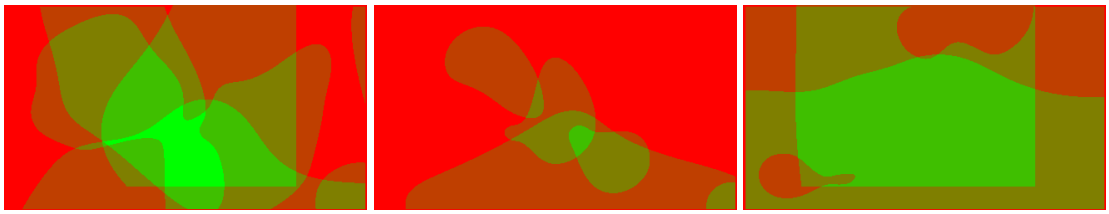


(a) CINR, Excess (b) CINR, Map (c) CINR, K-Var

(d) ESNR6, Excess (e) ESNR6, Map (f) ESNR6, K-Var

(g) ESNR54, Excess (h) ESNR54, Map (i) ESNR54, K-Var

Figure 6.17: Kriged maps for combined CU WiMax measurements using the CINR metric.



(a) CINR (b) ESNR6 (c) ESNR54

Figure 6.18: Boolean threshold-based maps for map-combined CU WiMax measurements. The CINR map uses a threshold of 40 dB and the ESNR maps use a threshold of 20 dB.

6.3 Case Study: Verizon LTE Coverage on the University of Colorado Campus

In this second case-study, the abilities of the geostatistical coverage mapping methodology are applied to a network utilizing a similar technology, however at a very different frequency: 700 MHz. In particular, the aim is to map the coverage of the Verizon Wireless Third Generation Partnership Project (3GPP) LTE network over the extent of the CU campus. LTE is a Fourth Generation (4G) compatible protocol that is also backwards-compatible with earlier Third Generation (3G) networks. LTE is favored by many to be the target of the next major cell technology roll out in the United States, and as such, is an especially interesting technology to study [186]. Verizon has deployed an early prototype LTE network operating around 700 MHz in central Boulder, Colorado, which allows the fairly unique analysis presented here. Table 6.3 shows the location and specifications of the LTE BSs [240].

6.3.1 Measurement Apparatus and Procedure

For the sake of consistency and comparability, measurements are made as before on a triangular lattice with a constant lag of 100 m. In addition, in this measurement campaign more care is paid to making systematic cluster measurements. Although the general measurement goals remain the same as in the previous study, the measurement apparatus and measurement procedure require some substantial adjustment for the LTE data collection. Figures 6.19 and 6.20 show the design of the LTE measurement apparatus. At the core of the measurement rig is a laptop computer running Windows XP and the JDS Uniphase (JDSU) E6474A network analysis and drive test software [53] connected to an LG VL600 LTE USB dongle. The JDSU software interfaces with the device to connect to the Verizon LTE network and collect real-time data about network performance.

As before, a hardcopy map is used to locate measurement points on a best-effort basis. Unlike the WiMax measurements, in this measurement campaign active measurements of upsteam and downstream throughput are collected at each point as well. To accomplish this, the program iperf is used to perform a 60 second bidirectional Transmit Control Protocol (TCP) throughput test. The general measurement procedure is given in algorithm 4.

Site	Sector	Latitude	Longitude	Site Name	Azimuth	PCI Grp ID	PCI ID	OTA PCI	N Obs.
85	1	40.017486	-105.252212	Central Boulder	10	72	0	216	0
85	2	40.017486	-105.252212	Central Boulder	95	72	1	217	0
85	3	40.017486	-105.252212	Central Boulder	180	72	2	218	2
119	1	39.997778	-105.2615	CU Campus	335	78	0	234	42
119	2	39.997778	-105.2615	CU Campus	70	78	1	235	12
119	3	39.997778	-105.2615	CU Campus	180	78	2	236	0
138	1	40.004656	-105.260597	Buffs	290	123	0	369	60
138	3	40.004656	-105.260597	Buffs	205	123	2	371	55
294	1	39.984708	-105.233044	Table Mesa	340	80	0	240	0
294	2	39.984708	-105.233044	Table Mesa	110	80	1	241	0
294	3	39.984708	-105.233044	Table Mesa	240	80	2	242	0
391	1	40.007092	-105.276575	The Hill	350	75	0	225	22
391	2	40.007092	-105.276575	The Hill	90	75	1	226	30
391	3	40.007092	-105.276575	The Hill	220	75	2	227	0
392	1	40.016128	-105.259997	Walnut	0	73	0	219	0
392	2	40.016128	-105.259997	Walnut	180	73	1	220	27
392	3	40.016128	-105.259997	Walnut	270	73	2	221	25
492	1	40.018028	-105.277768	DT Boulder	0	74	0	222	0
492	2	40.018028	-105.277768	DT Boulder	160	74	1	223	33
492	3	40.018028	-105.277768	DT Boulder	260	74	2	224	0
644	1	39.995314	-105.233431	Baseline n Foot	0	71	0	213	0
644	2	39.995314	-105.233431	Baseline n Foot	90	71	1	214	0
644	3	39.995314	-105.233431	Baseline n Foot	270	71	2	215	0
650	1	40.007925	-105.268236	Whizzer	270	76	0	228	39
650	2	40.007925	-105.268236	Whizzer	90	76	1	229	31
650	3	40.007925	-105.268236	Whizzer	180	76	2	230	37
652	1	40.037164	-105.246139	Ara	270	59	0	177	0
652	2	40.037164	-105.246139	Ara	55	59	1	178	0
652	3	40.037164	-105.246139	Ara	180	59	2	179	0
694	1	40.008431	-105.2577	Bison	0	77	0	231	2
694	2	40.008431	-105.2577	Bison	120	77	1	232	1
694	3	40.008431	-105.2577	Bison	235	77	2	233	39

Table 6.3: Specification and location of Verizon LTE cell basestations around Boulder, Colorado. The column N Obs. provides a count of the number of times each BS was observed in our data collection.

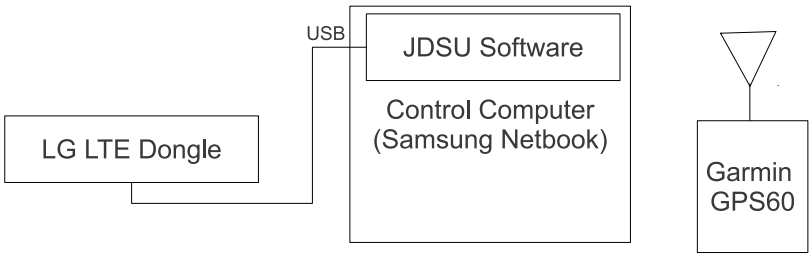


Figure 6.19: Diagram showing connectivity and specification of LTE measurement cart.

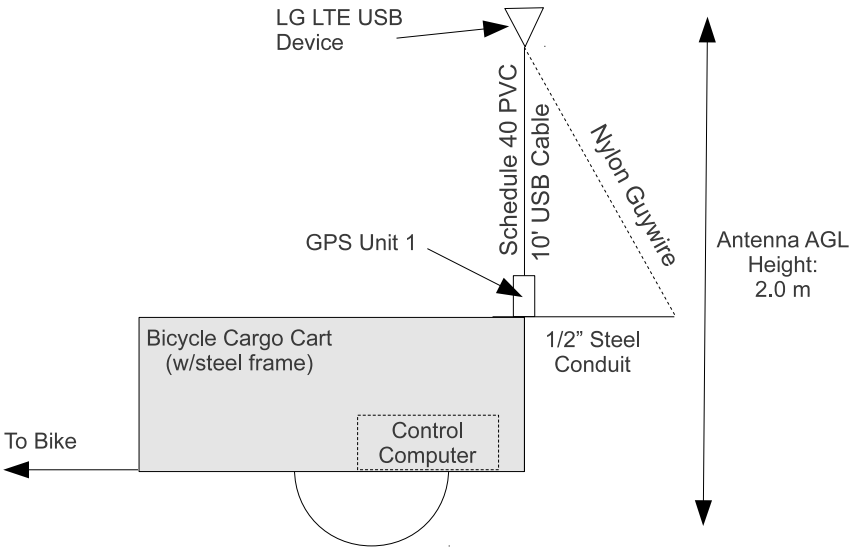


Figure 6.20: Diagram of LTE measurement cart.

Algorithm 4 Point testing procedure for LTE measurement

1: Locate a point to measure using the map. Try to get as close to the actual point as possible. If it is on/in a building or other obstacle, go to the nearest accessible outdoor location and do the measurement there instead. If the point is totally unreachable (i.e. closest point is 100m+ away), then circle the point and move on.
2: Write down the ID of the point being measured (i.e., 12 4 0) for the 0th measurement at point 12 4.
3: Using the hand held GPS, write down the current location and accuracy of the GPS position.
4: Using the computers clock (or a synchronized watch), write down the current time.
5: Click the green circle in the JDSU software to start a test.
6: After 5 seconds, a command window will open behind the main window and will run a throughput test. After 60 more seconds, this window will close.
7: Click the Stop button (or press Shift + F9) to stop the test.
8: Mark off the location just measured on the map with an X.
9: Every third point, perform 2 additional "cluster measurements" where you select a nearby point within an 8 meter radius of the original point (20 wavelengths at 700 MHz). These points should be more or less selected at random.
10: Move to the next point.

At each point, the JDSU software collects a continuous stream of measurements using a large number of metrics. This data is stored in a proprietary file format that can be exported to a parseable American Standard Code for Information Interchange (ASCII) Comma Separated Value (CSV) file after measurement is completed.

LTE measurement differs from WiMax measurement in an important way: the measurement device is a functional LTE radio and as such will connect to the best network available (ranking by Reference Signal Received Power (RSRP)) in order to pass traffic. A given measurement point may be served exclusively by a single cell, or by a combination of cells. Hence, some number of measurements are available for most cells that serve the CU campus, but as a result of this behavior, the strongest cells receive the most measurements. And, the resulting measurement set is more descriptive of the combined coverage of the cells, rather than a complete model of any particular BS.

6.3.2 Performance Metrics

One question this study seeks to answer is which metric is the best indicator of performance for an LTE network of this kind. To this end, all metrics that are available through the JDSU software are collected. As was done in section 6.1.2, an analysis is then performed to determine how well or poorly they predict application-layer performance. [111] descibes the physical-layer metrics that can be collected by the JDSU software and the 3GPP/LTE specification describes how many of these metrics are calculated [5]. The set of metrics collected is:

- Path Loss (PL): Path Loss in dB, calculated by the User Equipment (UE) using information from the BS.
- SNR: SNR in dB.
- RSSI: Three RSS measurements are available. It is not clear how they differ, so all three have been collected.
- Throughput: Upstream and downstream throughput are collected, however only an upstream

throughput test is performed, and the downstream channel is left unloaded. Throughput values are available at different layers, but are all simply a constant offset from one another.

- Transmit Power: Transmit power used by the UE, and the BS for the Sounding Reference Signal (SRS).
- Channel Quality Indicator (CQI): Values are available for the wide-band, sub-bands, and multiple code-words. All available are collected.
- Block Error Rate (BLER): Both Physical Downlink Shared Channel (PDSCH) and overall BLER are collected.

In order to understand how well the lower-level metrics predict the ultimate application-layer performance, an Analysis of Variance (AOV) is performed using the upstream throughput as the target and the physical-layer metrics as the factors to be tested. The results show that UE transmit power contributes most to higher-layer performance, and in fact, is inversely correlated with the performance, since the UE will choose a lower power when the upstream radio channel is good, and a higher power when it appears to be lossy. This factor produces an F-value of 385.6 and a p-value $<< 0.01$. Other significant factors include the PL (F-value = 90.6, p-value $<< 0.01$), SNR (F-value = 26.4, p-value $<< 0.01$), RSSI0 (F-value = 19.0, p-value $<< 0.01$), and distance from the BS (F-value = 19.5, p-value $<< 0.01$). Figure 6.21 shows the relationship between the best performing physical-layer performance metrics and upstream throughput.

It makes sense that the transmit power has a strong effect on the uplink throughput. And it stands to reason that PL would be a significant factor if the UE transmit power is, since the PL is used directly in the calculation of the transmit power. Similarly, the RSSI, SNR, and distance are all strongly correlated with the PL. However, this poses a problem: if the chosen transmit power is derived from a formula based on PL and CQI, and the transmit power affects the uplink throughput, then how can we know whether PL or CQI are themselves good predictors of channel quality, or whether they are being misused to tune the transmit power and hence bias the results towards or away from their values? Unfortunately, it is impossible to resolve this confound without performing modifications to the LTE protocol itself, and in this study we

have no control over the operators' infrastructure. Hence, this thesis proceeds with the assumption that the calculated PL (which is indeed determined using a channel-sounding mechanism) is the best performance predictor available, if it is not possible to make higher-layer performance tests directly. This is perhaps an unintuitive result, since PL does not consider non-flat fading accross subcarriers (as CQI does), but is nontheless the conclusion supported by the results.

A final question is how well correlated the various physical-layer metrics are *with each other*. Figure 6.22 shows the correlation between path loss and four other metrics. As expected, transmit power and RSSI appear to be strongly correlated with PL. This stands to reason because RSSI is likely calculated directly from path loss, and the UE transmit power is computed explicitly using path loss. The relationship between distance (from the connected BS) and SNR are less trivial, but there is still a clear and statistically significant correlation in both cases.

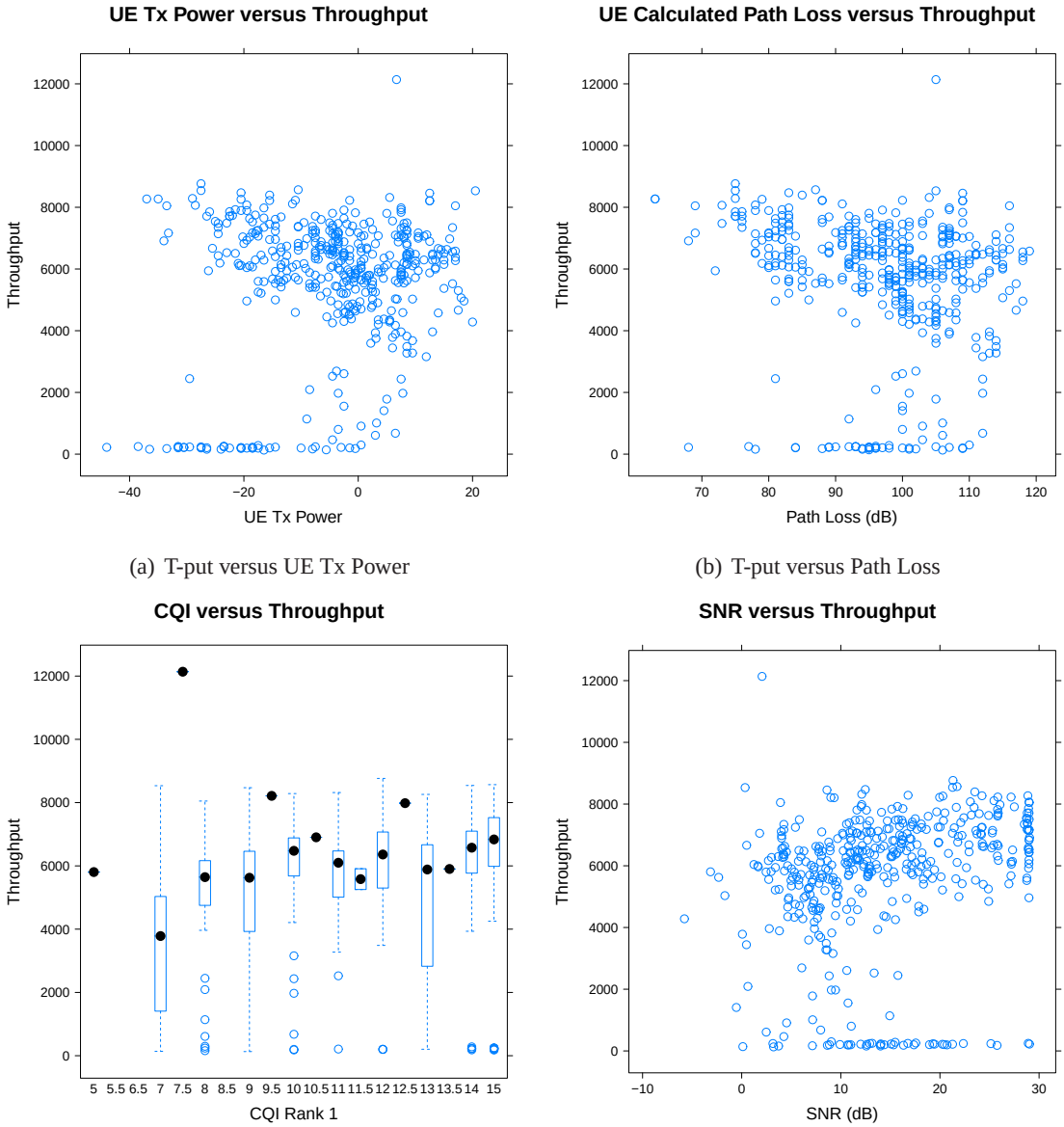
As could probably be inferred, the three RSS metrics are extremely similar, having a pearson correlation coefficient of between 0.97 (for RSSI1 versus RSSI2) and 0.99 (for RSSI1 versus RSSI0). The differences between these RSS values is dominated by a small location shift, supporting the theory that these are measurements of the same signal using different methods (or hardware). As such, use of any of the three RSS values produced by the measurement software should be sufficient to model RSS. In fact, any two can be used to predict the third using a simple linear mixture model:

$$RSSI_0 = 0.38RSSI_1 + 0.62RSSI_2 + 7.13 \tag{6.1}$$

This model has a residual standard error of 1.5 dB, an R^2 of 0.98, and an F-statistic of 11,720. Similarly, the PL metric and the RSSI metrics are closely related:

$$RSSI_0 = 41.38 - 0.96PL \tag{6.2}$$

which implies that the EIRP of the BSs is approximately 41 dB. This fit has a residual standard error of 1.66, an R^2 of 0.98 and a F-statistic of 19,430 indicating an exceptionally tight fit.



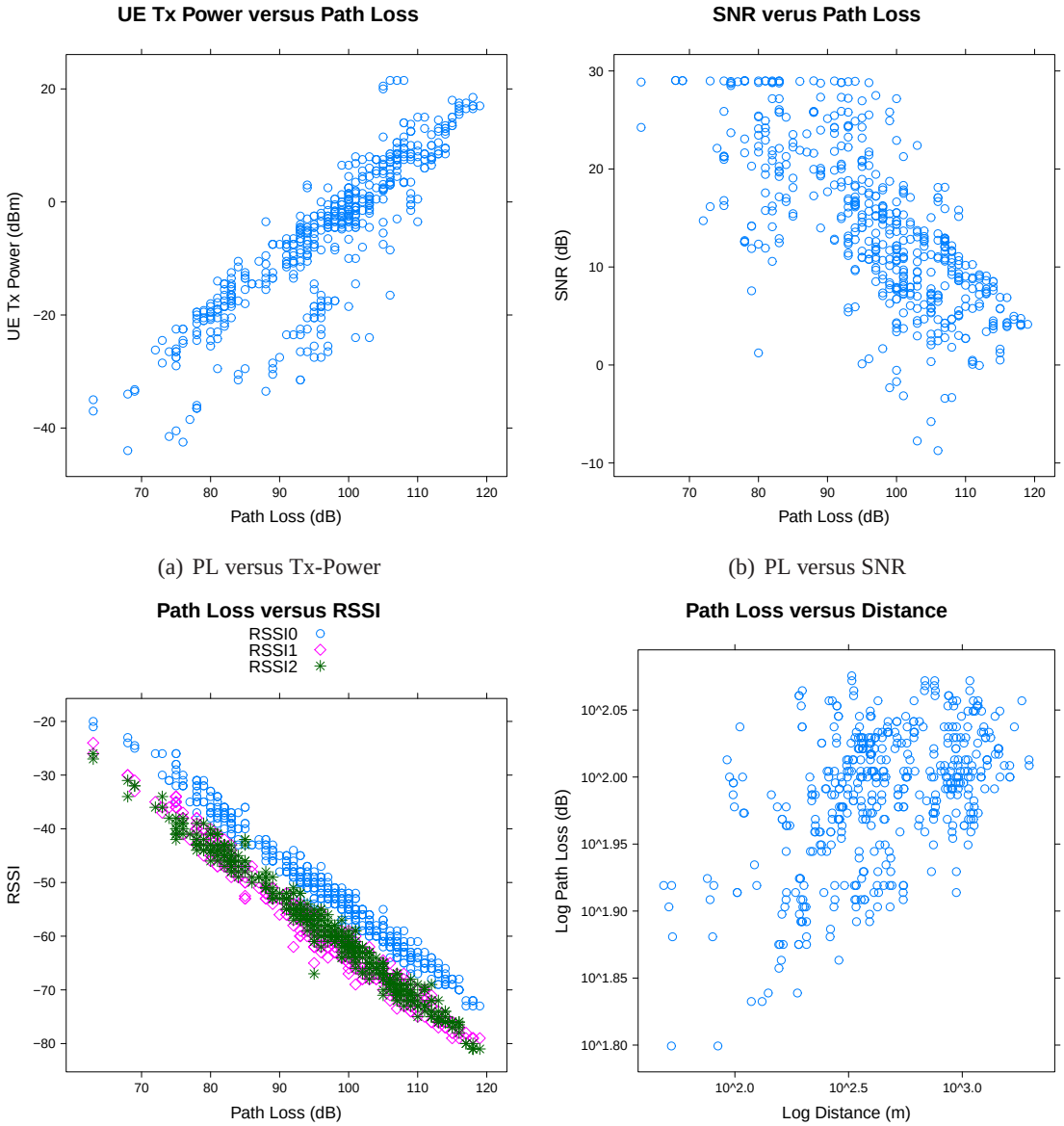
(a) T-put versus UE Tx Power

(b) T-put versus Path Loss

(c) T-put versus CQI

(d) T-put versus SNR

Figure 6.21: Correlation between upstream throughput performance and physical-layer metrics.



(a) PL versus Tx-Power

(b) PL versus SNR

(c) PL versus RSSI

(d) PL versus Distance

Figure 6.22: Correlation between physical-layer metrics.

6.3.3 Spatial Data Characterization and Variogram Fitting

In order to allow for comparability and to produce a more easily solveable Kriging matrix, the throughput measurements are normalized as such:

$$T_{kbps} = (T_{kbps} - min(T_{kbps}))/(max(T_{kbps}) - min(T_{kbps})) \tag{6.3}$$

The result of this transformation is that the throughput value at any given point is actually the fraction of total observed throughput seen at that point.

Figures 6.24 and 6.25 show the LTE measurements for all BSs and for the most observed BS, 369, for each of the seven metrics. The combined measurements and the measurements for 369 appear to have a similar distributional shape and to some extent the measurements for 369 appear representative of the measurements as a whole. Figures 6.26 and 6.27 show the variogram fits for this data, and figure 6.23 shows the log/log fit of path loss versus distance. The fits for the combined data are much better, presumably due to the uneven sampling of the single-BS data due to the way BSs are selected for measurement by the measurement apparatus. This further confirms that for these LTE measurements, the combined measurement map is most meaningful. Tables 6.4 and and 6.4 provide the fitted parameters for the combined data and a subset of interesting BSs.

6.3.4 Mapping with Ordinary Kriging

Ordinary Kriging proceeds as before, using the variogram models from section 6.3.3. For each combination of models and maps, an interpolated map is produced by Kriging the value at each pixel. Figures 6.28 and 6.29 show the final Kriged maps for the seven metrics for both BS 369 and all measurements combined. In the case of the maps in 6.28, the PL map shows a clear location and direction of transmission for the the BS. The remaining physical-layer metrics provide a consistent picture of the abilities of this particular node, whose range of functional coverage is actually fairly limited. The application-layer (i.e., throughput) metrics are clear outliers. As can be seen in the plot of measurements in figure 6.25, upstream and downstream throughput tend to largely disagree—at spots where upstream throughput is at its peak, downstream

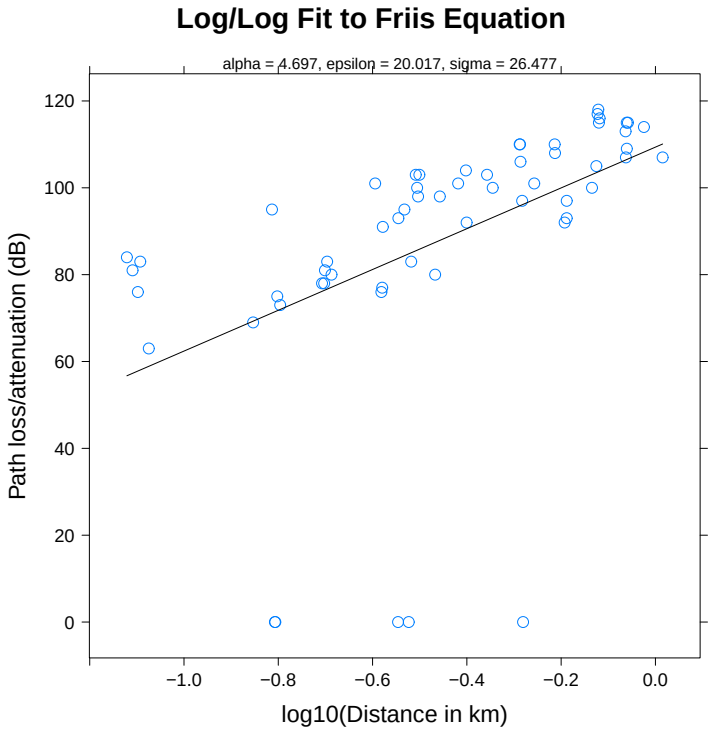
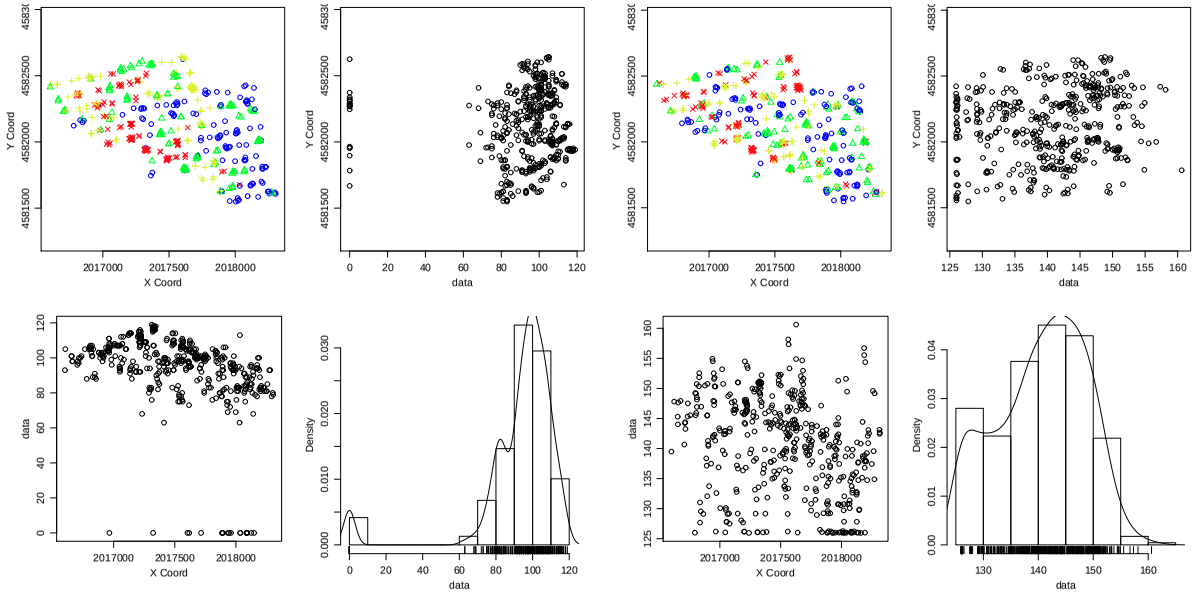
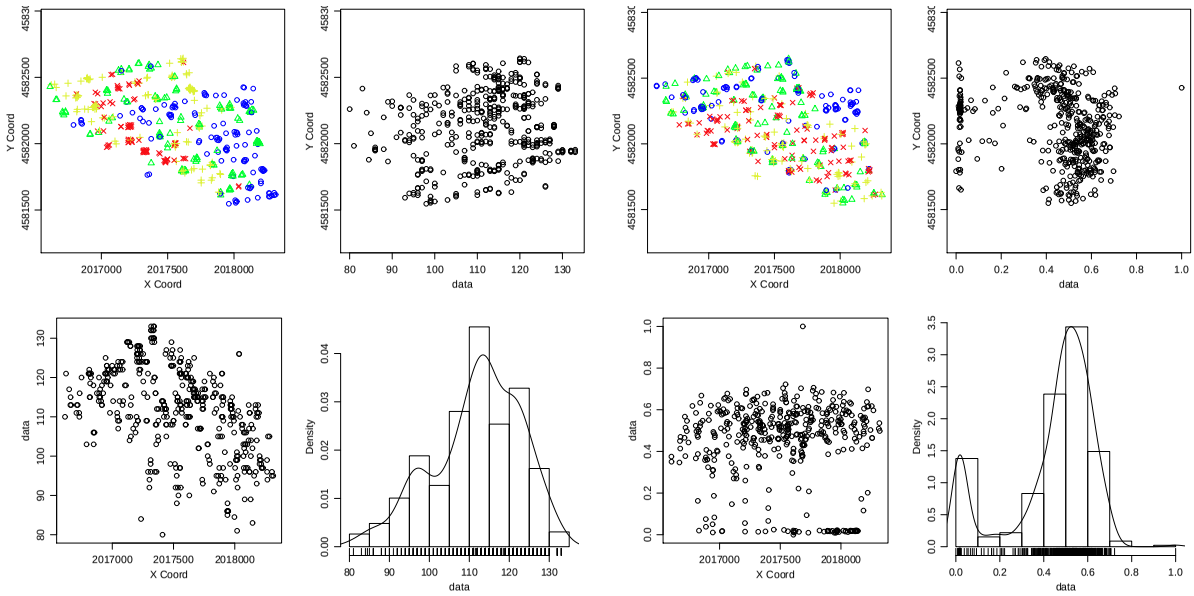


Figure 6.23: Log/log fit of path loss for LTE BS 369.

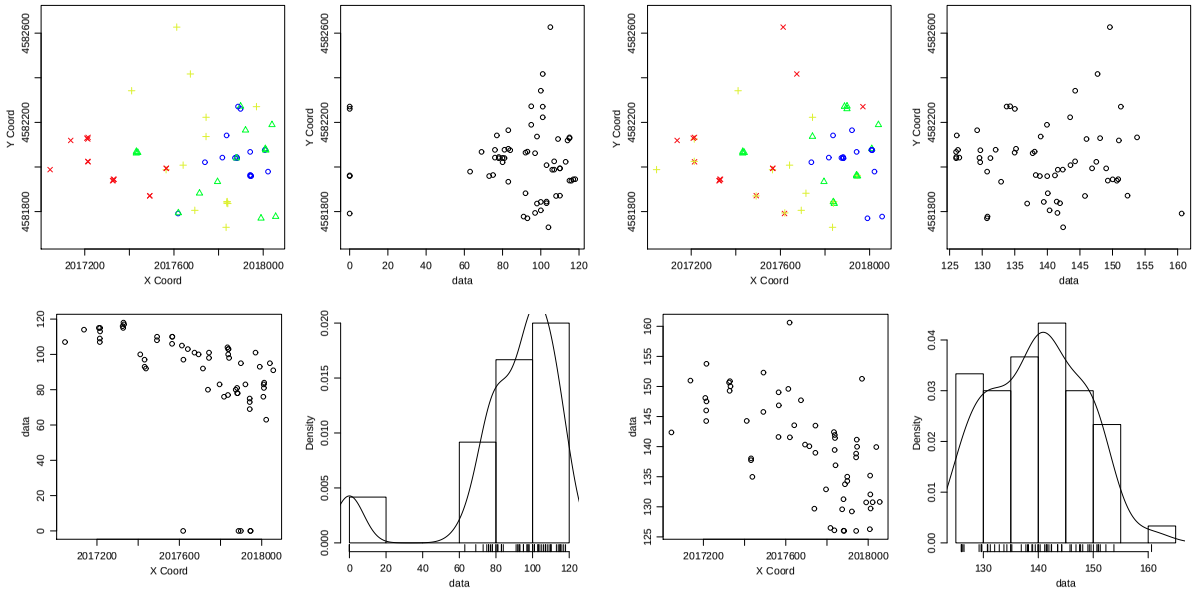


(a) PL (b) SNR

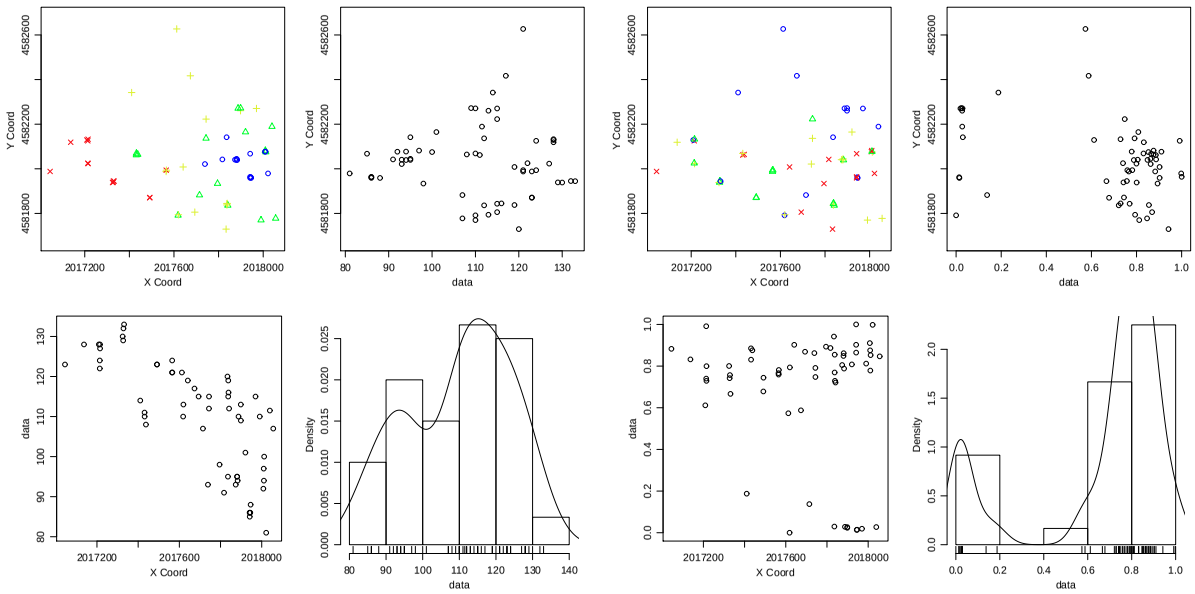


(c) RSS0 (d) UTPUT

Figure 6.24: LTE Measurements.

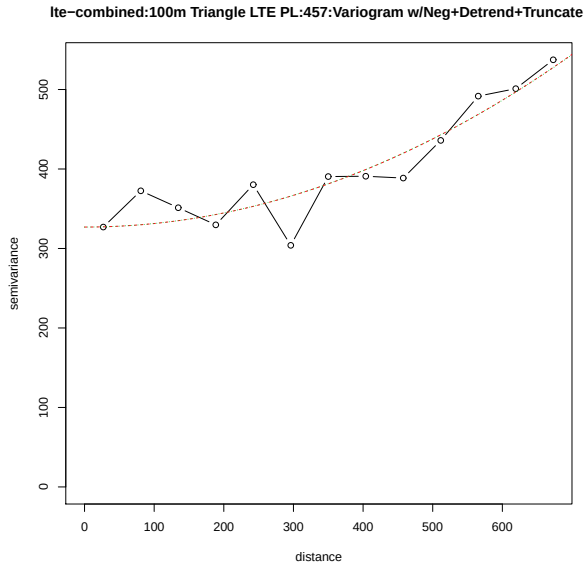


(a) PL (b) SNR

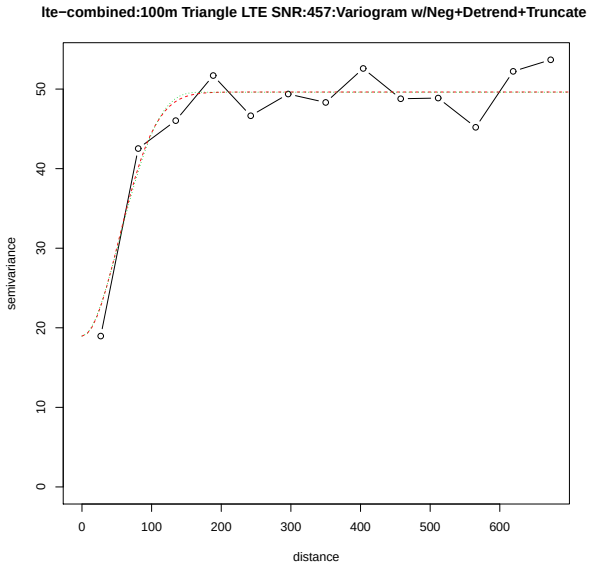


(c) RSS0 (d) UTPUT

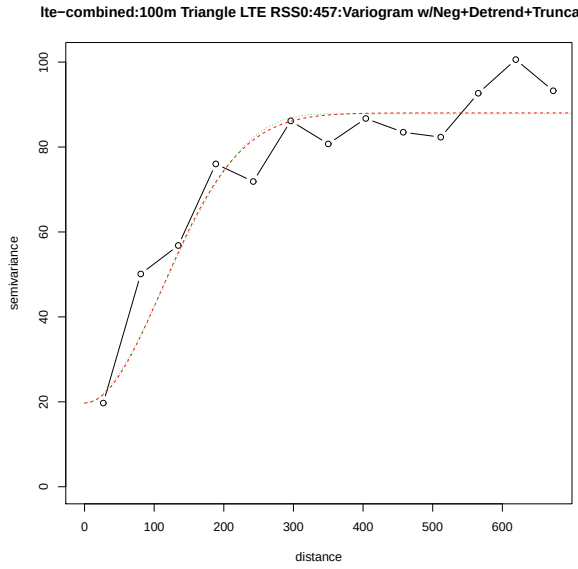
Figure 6.25: Measurements for LTE BS 369.



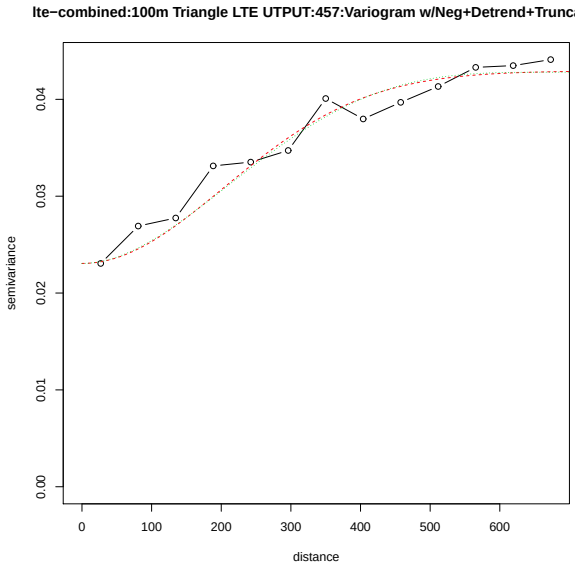
(a) PL



(b) SNR

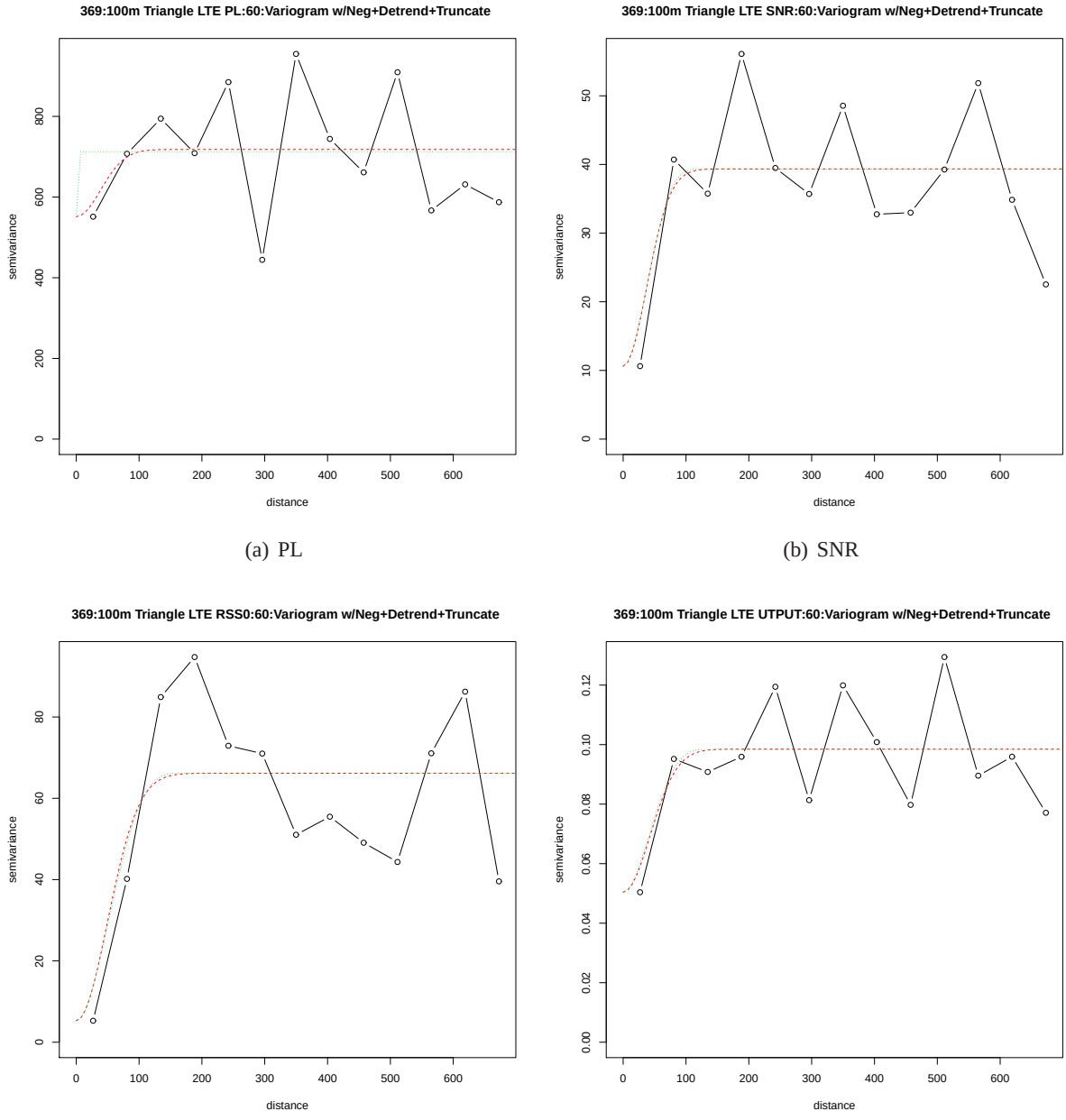


(c) RSS0



(d) UTPUT

Figure 6.26: Variogram fits for all measurements combined.



(a) PL

(b) SNR

(c) RSS0

(d) UTPUT

Figure 6.27: Variogram fits for BS 369.

AP Mac	Dataset	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
220	100lteRSS0	cubic	421.14	14.35	16.87	27	FALSE/FALSE	4.58	4.05	0.97
220	100lteSNR	cubic	190.48	18.52	13.98	27	FALSE/FALSE	5.34	5.18	0.43
220	100lteUTPUT	Gaussian	176.34	0.06	0.04	27	TRUE/TRUE	0.29	0.25	-0.29
221	100ltePL	cubic	127.49	10.89	24.83	25	FALSE/FALSE	4.60	4.62	1.36
221	100lteRSS0	Gaussian	61.98	8.86	18.57	25	TRUE/TRUE	4.17	3.86	1.06
221	100lteSNR	Gaussian	0.00	10.61	18.60	25	FALSE/TRUE	5.28	3.89	0.24
223	100lteUTPUT	cubic	129.50	0.02	0.03	33	TRUE/FALSE	0.19	0.19	-0.19
225	100ltePL	cubic	860.53	2.55	9.67	22	TRUE/TRUE	1.84	2.02	0.53
225	100lteRSS0	Gaussian	194.24	1.82	4.21	22	TRUE/FALSE	1.64	1.71	0.48
225	100lteSNR	Gaussian	219.27	9.89	16.55	22	TRUE/FALSE	3.46	3.65	0.96
226	100lteRSS0	cubic	167.10	5.27	6.94	30	TRUE/TRUE	3.01	1.93	0.65
226	100lteSNR	cubic	323.82	10.61	46.58	30	FALSE/FALSE	4.64	3.65	2.52
226	100lteUTPUT	Gaussian	107.85	0.04	0.06	30	FALSE/TRUE	0.26	0.24	-0.26
228	100ltePL	cubic	106.64	9.85	85.51	39	TRUE/TRUE	6.89	6.01	2.72
228	100lteRSS0	cubic	105.90	8.12	80.15	39	TRUE/FALSE	6.08	6.78	3.18
228	100lteSNR	cubic	174.24	17.83	66.97	39	TRUE/FALSE	6.86	5.00	2.05
228	100lteUTPUT	Gaussian	191.37	0.02	0.05	39	FALSE/FALSE	0.17	0.19	-0.17
229	100ltePL	cubic	136.71	2.50	88.24	31	TRUE/TRUE	6.04	4.13	3.37
229	100lteRSS0	Gaussian	54.56	2.00	84.81	31	FALSE/TRUE	6.19	4.58	3.02
229	100lteSNR	cubic	146.18	3.02	43.50	31	FALSE/TRUE	4.58	3.29	2.16
229	100lteUTPUT	Gaussian	55.35	0.01	0.03	31	TRUE/TRUE	0.13	0.12	-0.13
230	100ltePL	cubic	96234.59	14.57	2743822.04	37	TRUE/TRUE	4.07	12.90	15.56
230	100lteRSS0	Gaussian	97.17	10.02	99.68	37	TRUE/TRUE	6.42	5.19	3.73
230	100lteSNR	Gaussian	74.13	11.24	31.12	37	TRUE/FALSE	5.32	5.07	1.10
230	100lteUTPUT	Gaussian	40389.84	0.04	312.68	37	FALSE/FALSE	0.20	0.19	-0.20
233	100ltePL	cubic	42.48	1.00	1014.75	39	FALSE/TRUE	23.36	19.92	8.88
233	100lteRSS0	cubic	118.82	2.50	45.15	39	TRUE/FALSE	4.34	4.82	2.46
233	100lteSNR	Gaussian	47.05	4.06	30.42	39	TRUE/FALSE	4.36	3.62	1.34
233	100lteUTPUT	Gaussian	146.82	0.01	0.18	39	FALSE/FALSE	0.19	0.17	-0.19

Table 6.4: Table 1 of best fit statistics for variogram fitting of LTE data.

AP Mac	Dataset	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
234	100ltePL	cubic	343.61	1.25	68.95	42	FALSE/TRUE	3.65	6.32	4.39
234	100lteRSS0	cubic	387.17	9.25	57.03	42	FALSE/TRUE	4.51	5.11	3.25
234	100lteSNR	Gaussian	36.61	15.02	30.87	42	TRUE/TRUE	5.51	5.98	1.34
235	100ltePL	Gaussian	0.00	20.02	1.62	12	TRUE/FALSE	4.86	4.46	0.36
235	100lteRSS0	cubic	326.60	18.70	6.03	12	TRUE/FALSE	4.83	4.74	1.19
235	100lteSNR	cubic	47.07	57.78	0.00	12	TRUE/FALSE	0.00	10.61	6.22
235	100lteUTPUT	cubic	47.07	0.09	0.00	12	TRUE/FALSE	0.00	0.54	0.00
369	100ltePL	Gaussian	0.00	599.99	89.53	60	FALSE/TRUE	26.48	21.92	-0.00
369	100lteRSS0	Gaussian	69.73	5.29	60.87	60	TRUE/FALSE	5.46	3.94	2.40
369	100lteSNR	cubic	178.08	18.11	19.76	60	FALSE/TRUE	5.38	4.52	0.75
369	100lteUTPUT	cubic	188.70	0.06	0.04	60	FALSE/TRUE	0.29	0.26	-0.29
371	100ltePL	cubic	996.66	313.27	213.95	55	FALSE/TRUE	18.53	11.82	1.23
371	100lteRSS0	Gaussian	118.86	7.76	68.13	55	FALSE/FALSE	4.56	4.50	3.84
371	100lteSNR	cubic	411.27	20.09	26.95	55	FALSE/TRUE	5.05	4.57	1.50
371	100lteUTPUT	cubic	1450.56	0.05	0.06	55	FALSE/FALSE	0.23	0.24	-0.23
lte-combined	100ltePL	cubic	87330.05	326.96	486883.61	457	TRUE/TRUE	18.16	17.81	3.16
lte-combined	100lteSNR	Gaussian	74.62	18.96	30.67	457	TRUE/TRUE	5.09	4.30	2.12

Table 6.5: Table 2 of best fit statistics for variogram fitting of LTE data.

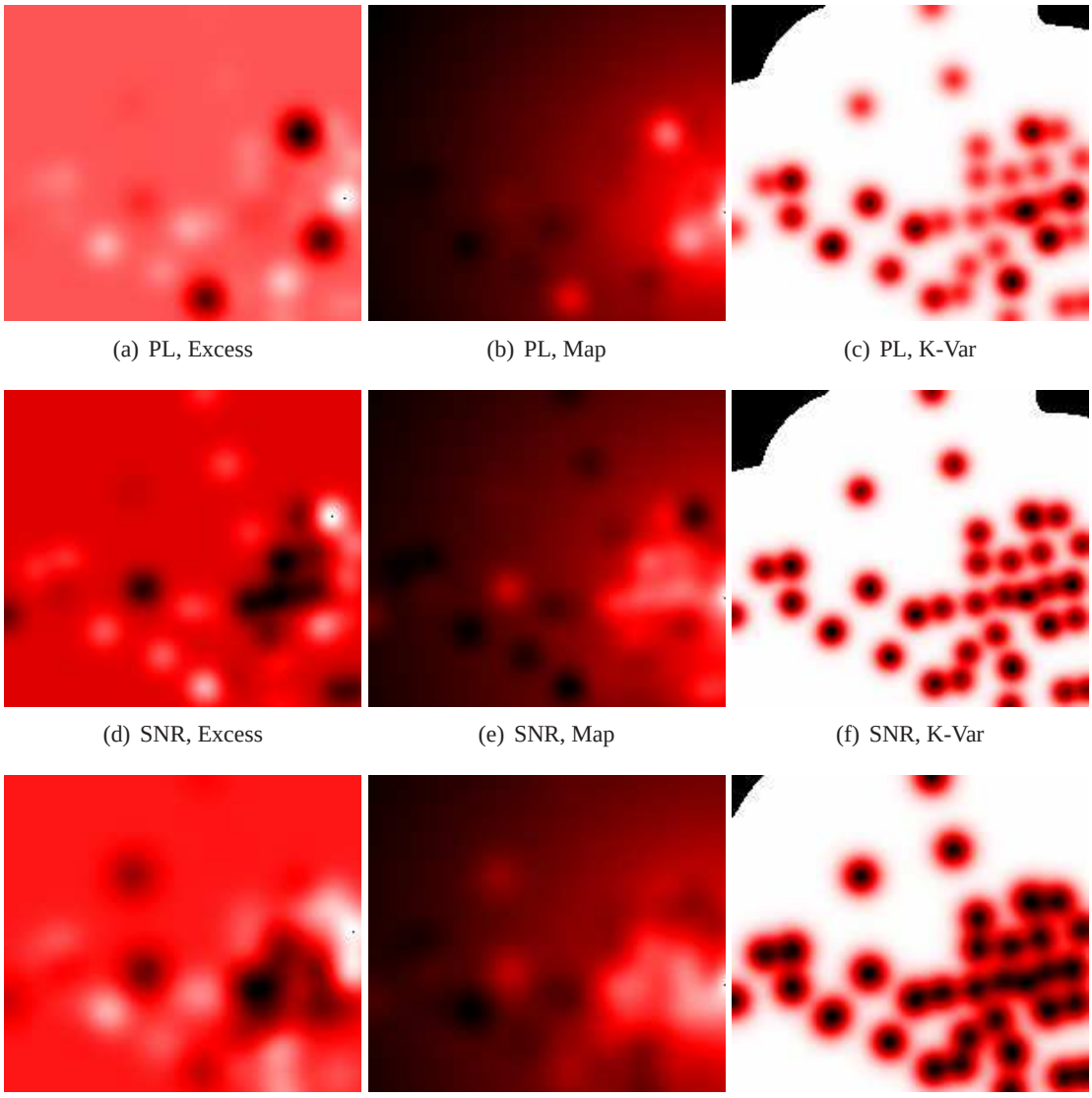
throughput may be zero. The interpolated maps reflect this: upstream throughput interpolates to a largely positive value with holes around regions where traffic could not pass, whereas downstream throughput predicts mostly a bitrate of zero, excepting those few regions where downstream tests were successful. It is likely the case that these results are a function of a problem in testing throughput further up the network, either via traffic rate limiting or outright blocking. However, we do not have enough control over the Verizon network to differentiate possible sources of error here. Instead, we think this helps to further highlight the fickle nature of higher-layer tests.

Figure 6.29 shows the map with all measurements combined and table 6.5 provides the performance results for these fits. Although the differences in the measurement procedure results in fewer co-located measurements with radically different values (i.e., only the best BSs should receive measurements), the combined mapping process is still somewhat unpredictable as the interpolation must resolve unrealistically large differences in value between nearby points. This is especially visable in the K-Var plots, which show that variance is minimal only immediately around the points and then grows rapidly moving away from each measurement. A better map might be obtained if the UE hardware were able to collect indepent measurements of each BS at each point (as was done in the WiMax tests). However, this isn't possible with the hardware used here and hence, there are some necessary limitations in terms of the resulting map. Were the data collected by Verizon themselves, it would be easy to address this limitation.

finally, figures 6.30 and 6.31 provide the map-combined maps for these measurements. The former is maximum-combined, and shows a very complex landscape for each metric. The latter provides threshold-combined maps using an SNR of 20 dB (or equivalent). The right-hand side shows which points are covered by this metric, and the left-hand side shows how many separate BSs cover the region with performance at least meeting the threshold value.

6.4 Modeling Small Scale Effects and Deviation from Stationarity

This section asks the question: how do measurements vary over small time scales and small distances? An underlying assumption of the Kriging process is that the process being modeled is stationary, meaning that the (fitted) mean is constant in both time and space. Clearly, this is a strong assumption that the (often

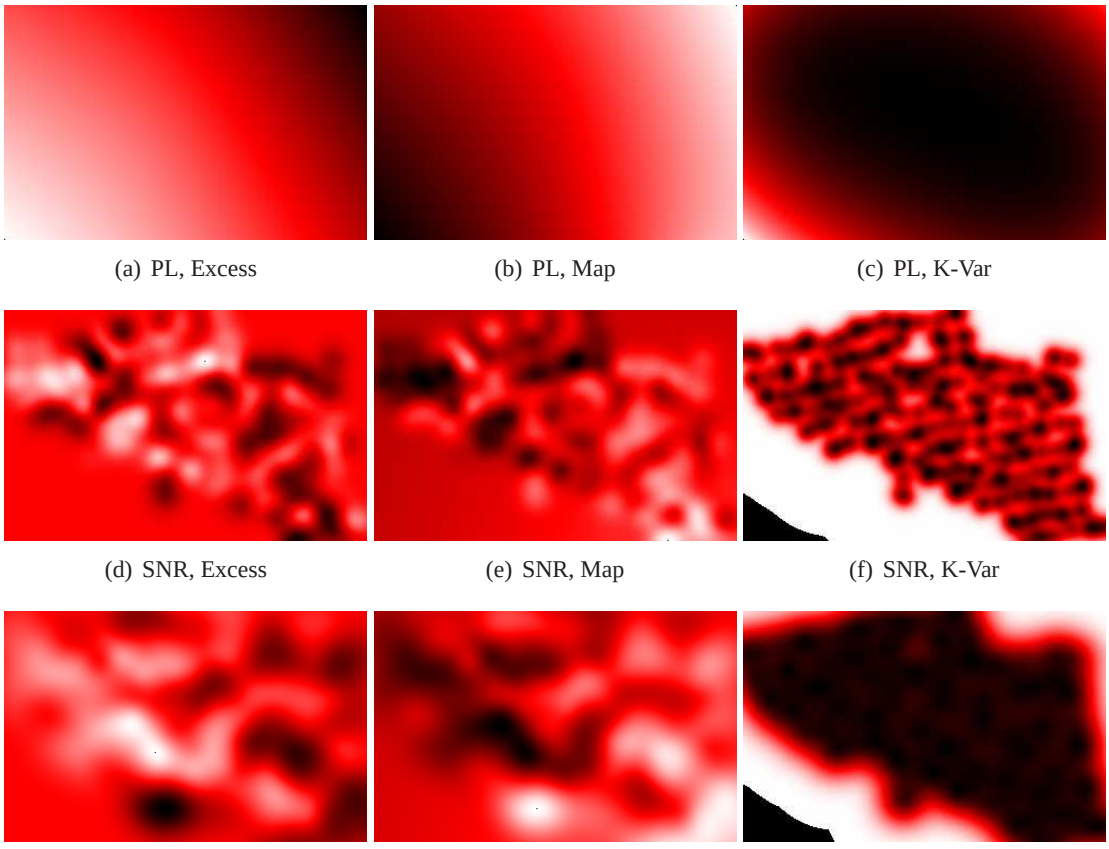


(a) PL, Excess (b) PL, Map (c) PL, K-Var

(d) SNR, Excess (e) SNR, Map (f) SNR, K-Var

(g) RSS0, Excess (h) RSS0, Map (i) RSS0, K-Var

Figure 6.28: Kriged maps for BS 369.

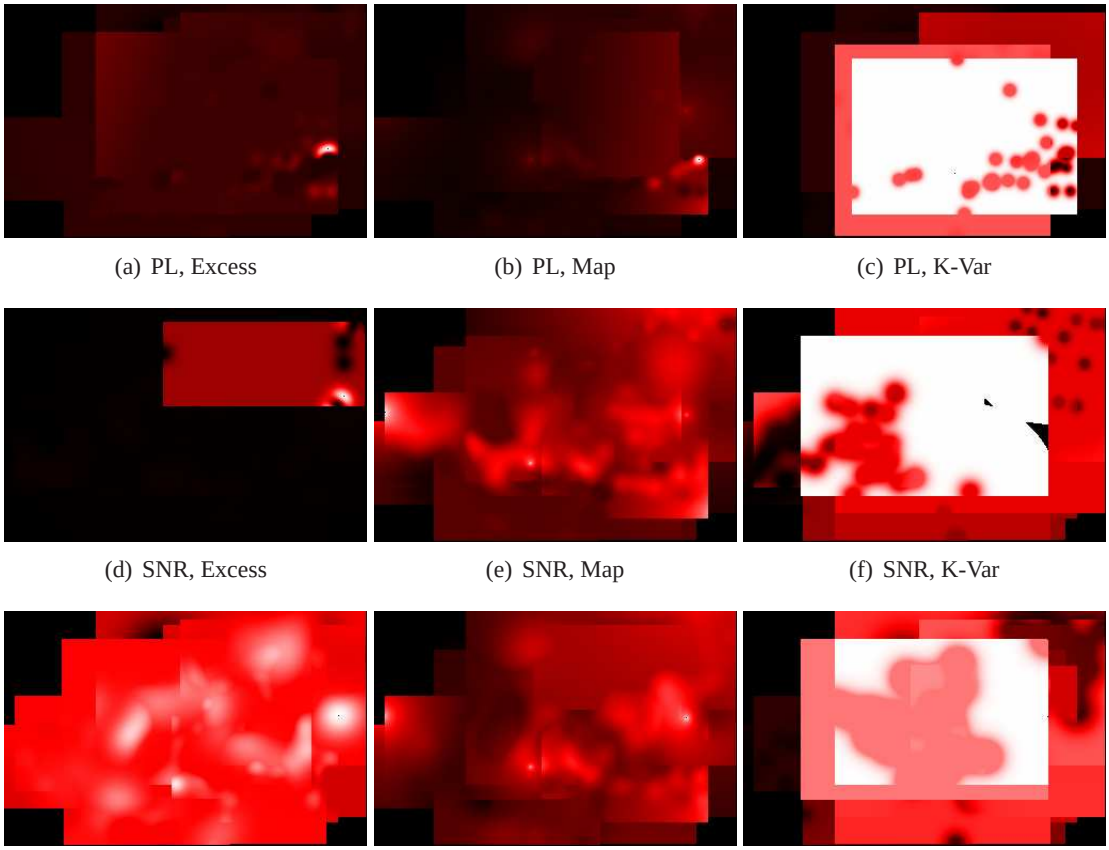


(a) PL, Excess (b) PL, Map (c) PL, K-Var

(d) SNR, Excess (e) SNR, Map (f) SNR, K-Var

(g) RSS0, Excess (h) RSS0, Map (i) RSS0, K-Var

Figure 6.29: Kriged maps for all measurements combined.

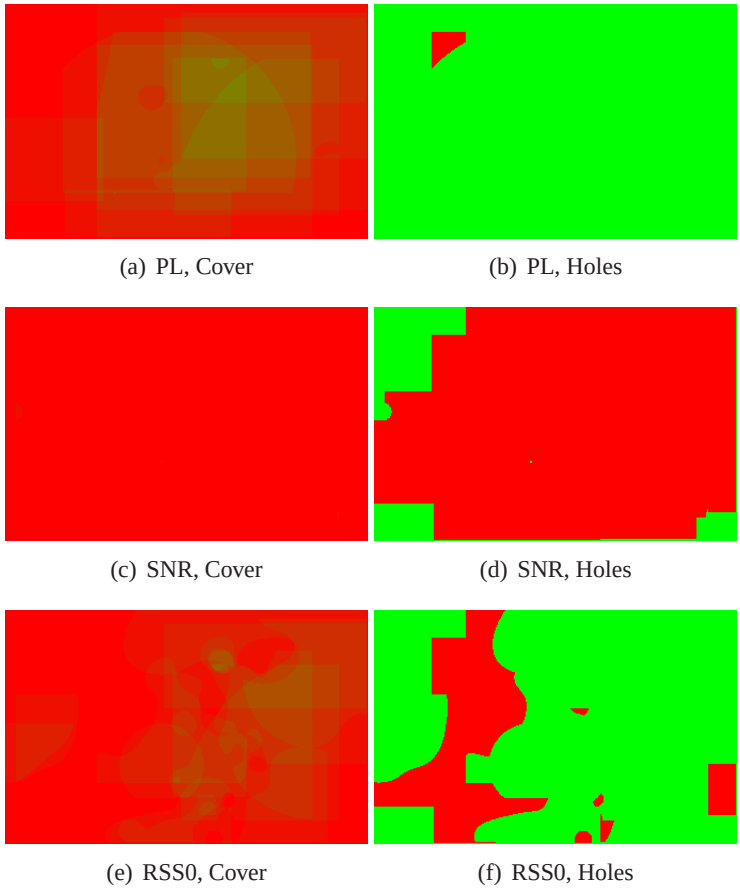


(a) PL, Excess (b) PL, Map (c) PL, K-Var

(d) SNR, Excess (e) SNR, Map (f) SNR, K-Var

(g) RSS0, Excess (h) RSS0, Map (i) RSS0, K-Var

Figure 6.30: Map-combined maps for all measurements combined.



(a) PL, Cover (b) PL, Holes

(c) SNR, Cover (d) SNR, Holes

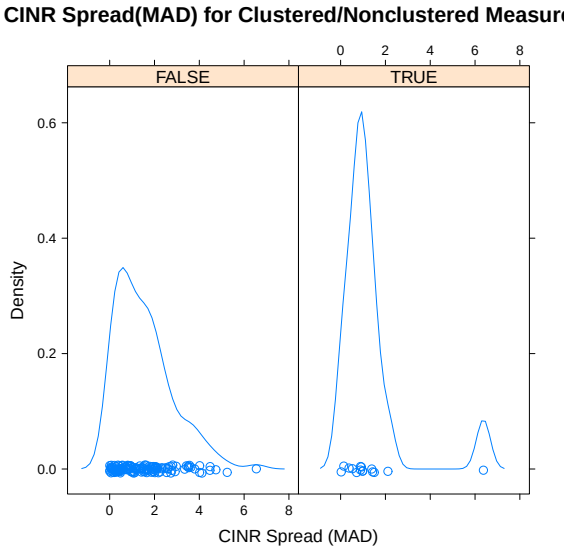
(e) RSS0, Cover (f) RSS0, Holes

Figure 6.31: Threshold-based map-combined maps for all measurements combined.

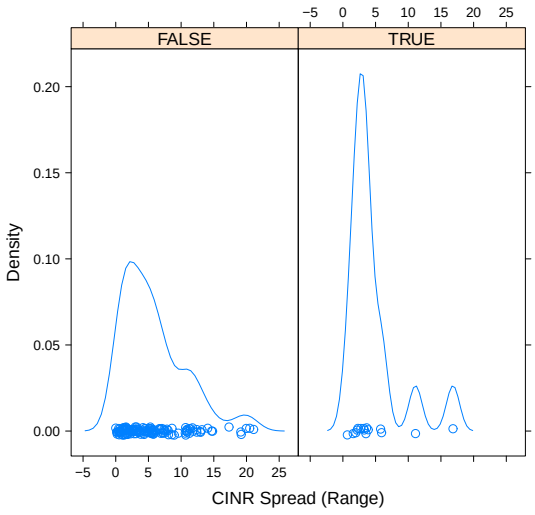
chaotic) radio environment is unlikely to obey. It is possible to loosen the stationarity assumption at the cost of substantial additional computational work, but in practice most users of Kriging processes opt to accept the implications of this assumption. By understanding how the radio environment changes over small time scales and small distances, a bound can be placed on repeated measurement variation and hence a bound on the implicit unavoidable error associated with the stationarity assumption.

As discussed in section 2.1, fading in the radio environment can be classified into small-scale and large-scale fades. Large-scale fades should be fairly constant over large distances and time, and hence are not troublesome—it is exactly the environment-specific large-scale fading effects that Kriging seeks to interpolate. However, small-scale fades can be highly varying in time and over small distances because they stem from multipath effects and (possibly mobile) scatterers. As a practical rule of thumb many experimenters average measurements within 40 wavelengths as a way to "average out" small-scale effects [132]. This section seeks to validate that standard practice as well as understand the scale of small-scale effects over short time scales.

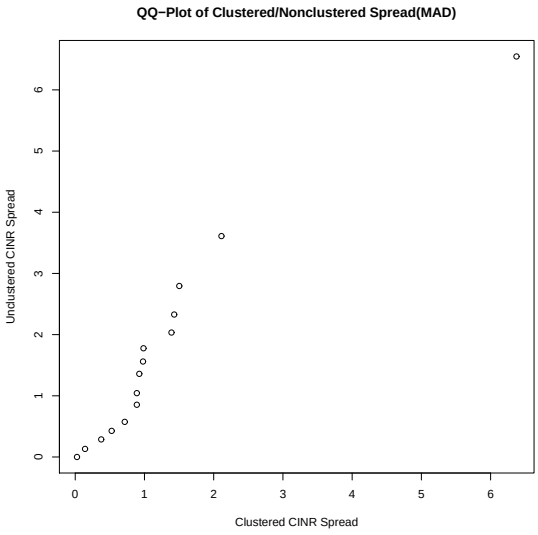
After the initial CU WiMax measurement campaign, a second campaign was undertaken to collect data at clustered locations so that the small-scale (in space and time) variation can be compared to large-scale trends. To this end, a random subset of approximately 15 grid points were selected and at each point three complete measurements were taken at random locations within 40 wavelengths of the original grid point. Figure 6.32 shows the amount of measurement spread observed at these closely clustered locations versus the amount of measurement spread between repeated measurements at the same location along with comparative QQ-plots of the distributions. Here, two different definitions of spread are used that make no assumptions about the underlying distribution of the data: range, which is simply the total difference between the largest and smallest observation, and Median Absolute Deviation (MAD), which is the median of absolute deviations from the data's median. Although the two distributions are not identical, they do appear to be Gaussian in both cases, with a similar central tendency and spread. Indeed, these distributions have been compared with a Welch two-sample t-test, two-sample Kolmogorov-Smirnov test, and Wilcoxan rank sum test (all of which test the null hypothesis that the difference in central tendency is significant), and none of the tests are willing to reject the null hypothesis that the data are drawn from the same distribution



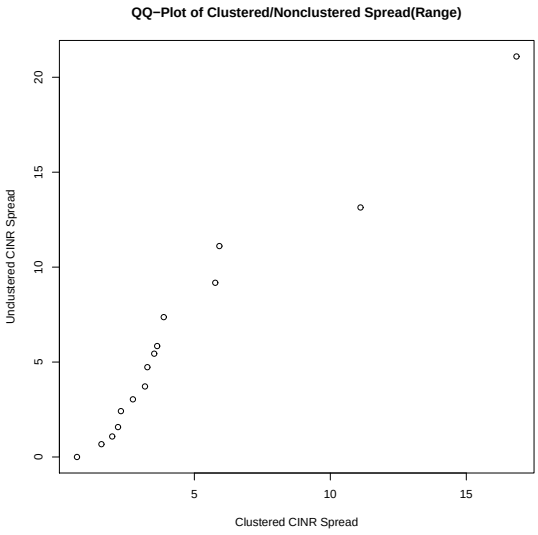
(a) Range PDFs



(b) MAD PDFs

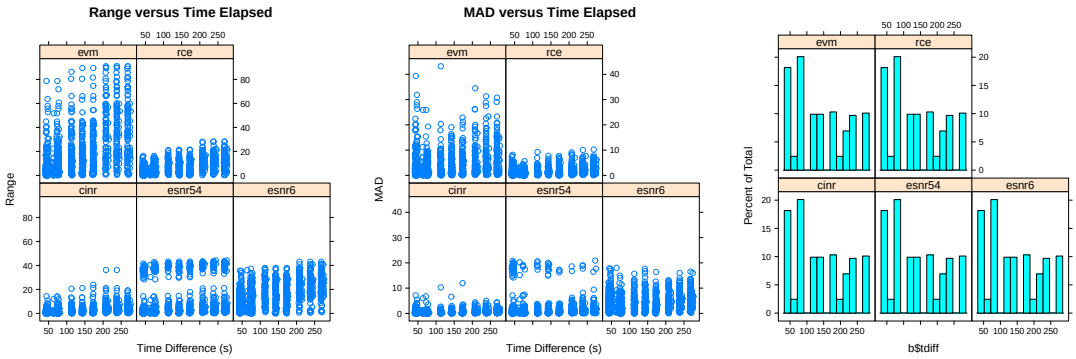


(c) Range QQ-plot



(d) MAD QQ-plot

Figure 6.32: Distribution of spread and comparative QQ-plots for measurements taken in within 40 wavelengths of each other (i.e., clustered) versus at the same point at different times (i.e., unclustered) for two different definitions of measurement spread (MAD and range).



(a) Range (b) MAD (c) Number of Samples

Figure 6.33: Amount of spread (variation) as a function of time elapsed between measurements using two different metrics.

(p-values are between 0.3 and 0.5). This result suggests that the amount of underlying small-scale fading in space can be sufficiently modeled by taking repeated measurements in time, at least in the environment studied here.

Figure 6.33 provides a different view of this intrinsic channel variability by plotting the amount of variation observed between repeated measurements taken at the same location as a function of time. This figure shows that the amount of variation appears to be fairly stable for all of the metrics over small time scales (several minutes). There is a slight increase in measurement spread observed for the RCE and EVM measurements, but this does not appear to be substantial, and may not be significant. Interestingly, the ESNR metrics appear to have more intrinsic variation than the simpler metrics, which may be due to the fact that these metrics take into account more degrees of freedom (i.e., independent fading on each subcarrier). Although it is likely that the radio environment is nonstationary at large time scales (days, weeks, and years), from these results it appears that the intrinsic variation is fairly stable on small time scales and hence a few repeated measurements are likely sufficient to characterize intrinsic variability.

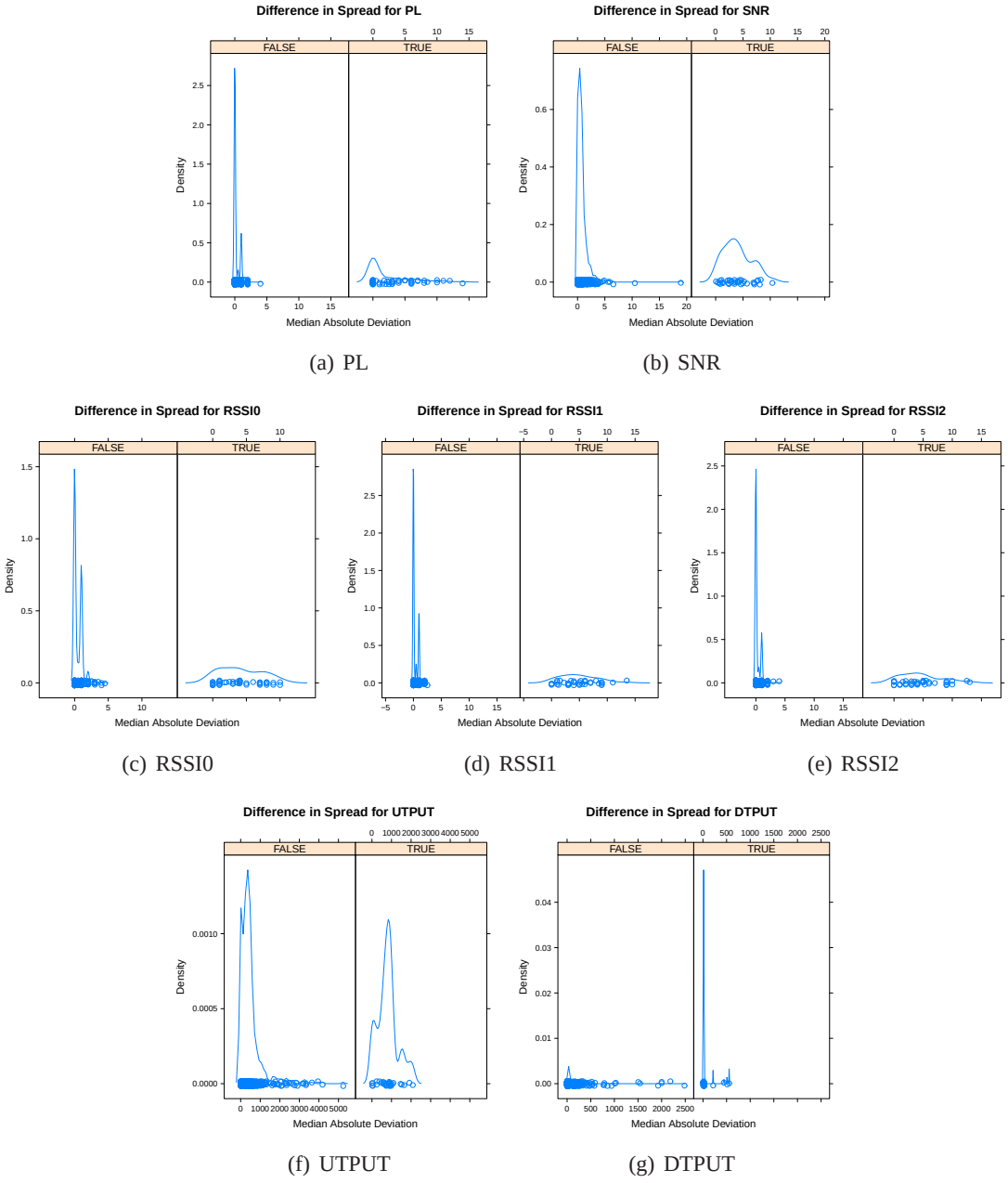
As a further investigation, clustered measurements were systematically collected at every third sample location in the LTE data collection process. Figure 6.34 shows the distribution of spread at repeated measurements in space (right column (TRUE), measurements clustered within 40 lambda) and repeated measurements in time alone (left column (FALSE)). The time-delayed repeated measures variance for the

physical-layer metrics and is centered around zero (the mean and median are zero for the three RSS metrics and PL and 0.57 forSNR). Consistent with the maps, the throughput metrics experience greater variation at baseline. The variation for clustered measurements appears to be somewhat larger. This result appears to disagree with the seemingly same-scale variation observed between clustered and time-delayed repeated measurements for the WiMax data. This is especially striking because the lower carrier frequency of LTE (700 MHz versus 2.5 GHz) would seem to imply a more stable channel. However, a deeper look at the distribution of this error reveals the the majority of greater variation is due to outliers. In fact, the amount of variation is only 4.2 dB for the PL metric at the 80th percentile (0 dB in the median). Similarly, the variation is approximately 7 dB at the 80th percentile ($\approx$ 4 dB at the median) for the RSS and SNR measurements. Still, this variation is larger than might be expected, and suggests two possible explanations:

- Because the WiMax measurements were made with a finely tuned and calibrated spectrum analyzer, and the LTE measurements were made with a COTSE USB radio, the measurement device is simply more noisy and this source of error is superimposed on the observed channel error.
- The spectrum is simply more variable at 700 Mhz than it is at 2500 MHz in this environment (i.e., more random attenuation from location-specific scatering).

Resolving which of these two explantions is correct is not possible with the current measurements and would require a comparative evaluation of measurement hardware that is outside the scope of this thesis research. Instead, the assumption is taken that the baseline repeated measures variance for LTE measurement and mapping is between 2 and 4 dB for clustered repeated measurements and approximately zero for time-delayed repeated measures. This indicates that unlike in the case of the WiMax measurements, multiple measurements at closely clustered locations may be necessary to sufficiently model the intrinsic small scale variation (in addition to or replacing multiple measurements in time).

Finally, consider figures 6.35 and 6.36, which show the amount of variation as a function of distance between repeated measurements (in time), and the number of measurements observed at different time delays. As with the the WiMax measurements which show little or no trend in small time scales, this data seems to confirm that the amount of time between repeated measurements will not substantially increase the



(a) PL (b) SNR

(c) RSSI0 (d) RSSI1 (e) RSSI2

(f) UTPUT (g) DTPUT

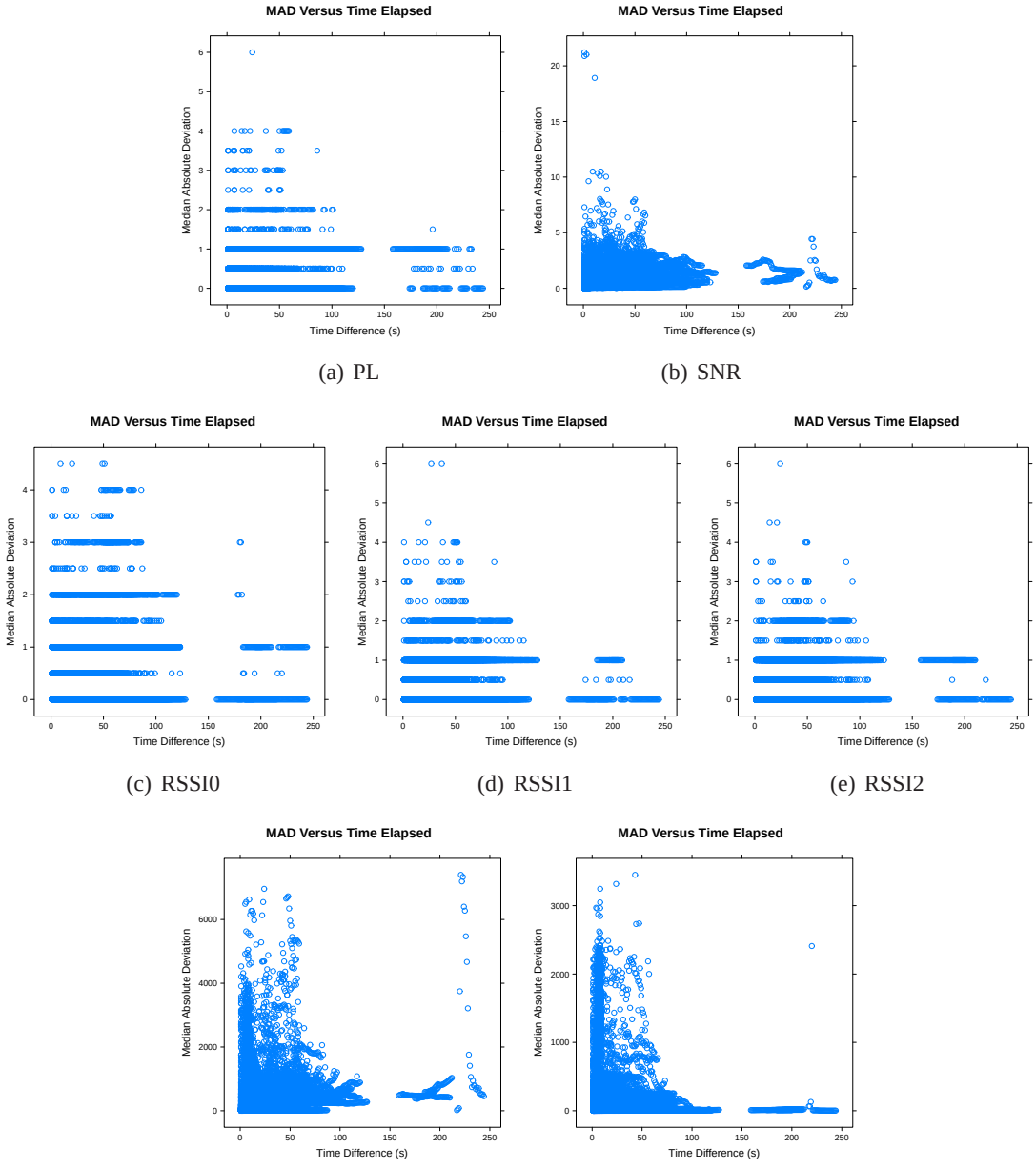
Figure 6.34: Comparison of mean absolute deviation for metrics both in the same place at different times (column FALSE, i.e. unclustered) and at different places and different times, but less than 40 wavelengths apart (column TRUE, i.e., clustered).

amount of variability observed. Instead, the number of measurements is a stronger predictor of the amount of observed variance, indicating that, in sum, a large number of co-located and clustered measurements may be necessary to model channel variability for LTE networks at 700 MHz. This includes, of course, the necessary reminder that hardware calibration (or lack thereof) may contribute (positively or negatively) to observed variance.

6.5 Summary and Conclusions

This chapter (in combination with chapter 5) has provided the first complete, real-world application of geostatistical modeling and interpolation to the problem of wireless coverage mapping. Although some other authors have proposed that geostatistical techniques may be appropriate for the domain, the work here is the first to actually apply the concepts and adapt them as necessary for the mapping of production networks. To analyze their efficacy, these techniques have been applied to two unique scenarios. This process has shown that robust coverage maps can be produced using a reasonably small amount of easily obtained data (several hundred samples on a hundred meter grid, for a space the size of a large university campus), which amounts to a tractable amount of routine “spade work” (approximately three days work for a single dedicated experimenter).

In the first scenario, extensive per-BS measurement of a 2.5 GHz WiMax network on the University of Colorado campus was performed. For that study specifically, this involves sampling on a 100m equilateral trianglular grid, removing trend, truncating measurements, and interpolating using Kriging. A custom measurement apparatus was developed to allow for measurement at arbitrary locations with high precision equipment. In the second scenario, coverage of a 700 MHz Verizon LTE network on the University of Colorado campus was tested. As compared to the WiMax measurement campaign, measurement methods were further refined to include additional clustered samples to permit better modeling of small-scale (in time and space) stochastic fading effects. In both scenarios the result is a set of maps whose meaning and confidence is well defined and whose accuracy is better than what could be expected with simple measurement-based methods (powerlaw fitting) or standard predictive models. In general, an error reduction of at least several dBs over *a priori* models is obtained.

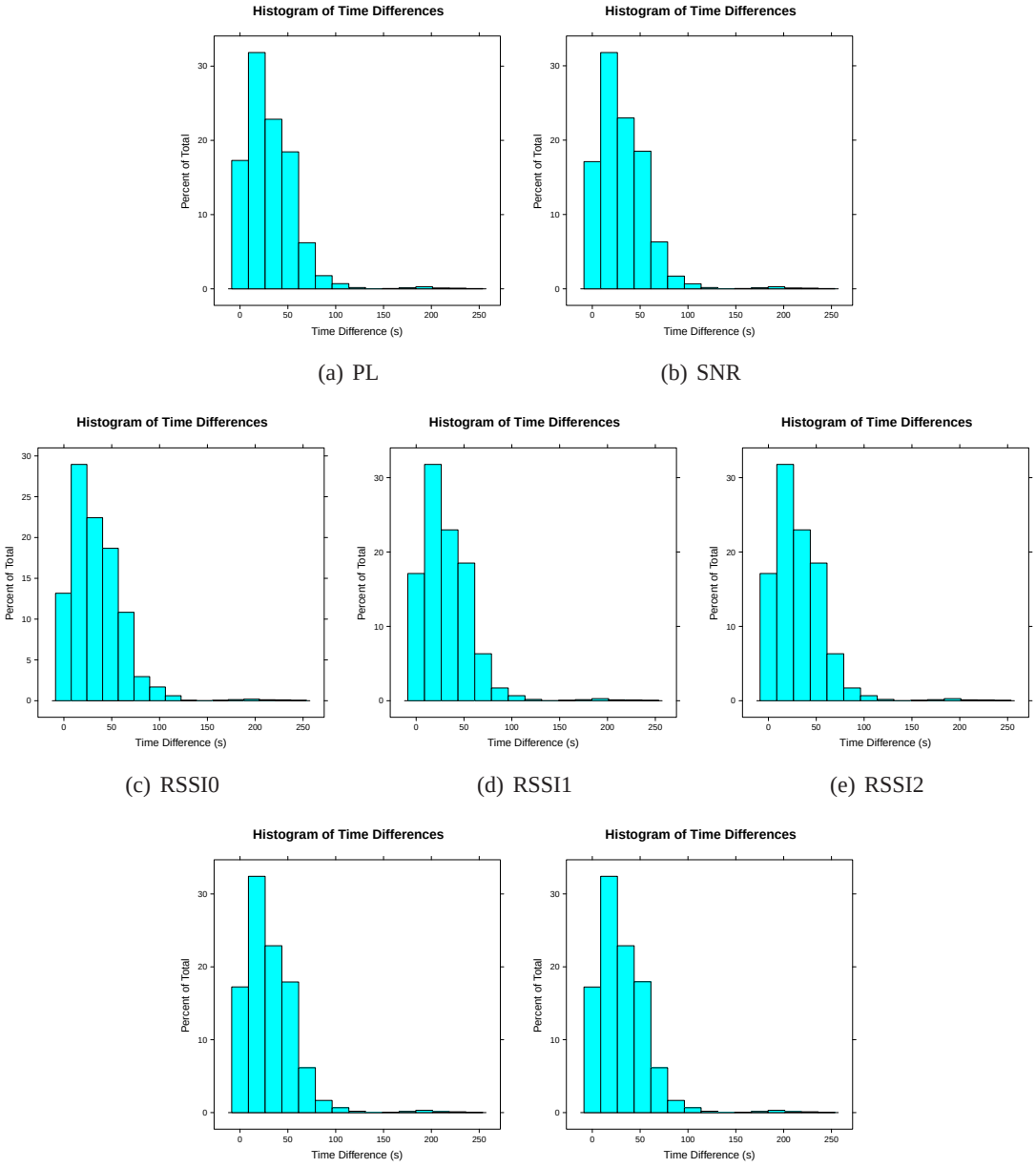


(a) PL (b) SNR

(c) RSSI0 (d) RSSI1 (e) RSSI2

(f) UTPUT (g) DTPUT

Figure 6.35: Mean absolute deviation as a function of time elapsed between measurements in seconds. There appears to be no discernable correlation.



(a) PL (b) SNR

(c) RSSI0 (d) RSSI1 (e) RSSI2

(f) UTPUT (g) DTPUT

Figure 6.36: Number of samples per bucket for time elapsed between repeated measurements.

The next chapter will provide additional case studies focussing on the performance of these methods in scenarios where measurements are collected less rigorously, but with greater ease.

Chapter 7

Drive-test and Crowd-Sourced Coverage Mapping

The previous chapter proposed a method for wireless network coverage mapping using geostatistical techniques and carefully selected measurements. This thesis argues that the most interesting and opportune area for coverage mapping is where the rubber meets the road: empicial, measurement-based techniques. However, it is not always possible or affordable to collect measurements on a uniform lattice. This chapter investigates two methods of more "convenient" sampling: drive-test measurements and crowd-sourced measurements. Drive-test measurements are the defacto method of measurement used by cell providers and many network planning consultants. These measurements are made by vehicles and are confined to city streets. While previous chapters have argued that this sort of measurement has clear sources of systematic bias, particularly from the wave-guiding effects of street-canyons, whether these measurements may still be useful for some general prediction is an open question this chapter will seek to address.

The second type of "convenience" sampling, is crowd-sourced measurements where a possibly large number of volunteers may choose to use their mobile devices to collect and contribute mesurements to a coverage mapping campaign. This approach has the pleasing attribute that crowd-based sampling will oversample the regions of highest interest (i.e., where people go) and undersample the areas of lowest interest (where they do not). Although this is still an emerging idea with a great deal of work needed to understand how well this may work, this chapter offers an initial step. First, a mobility model is used to simulate walking patterns and understand where measurements might be made in a system like this, and how often. These results show that crowd-sourced coverage mapping is likely feasible in terms of practical participation levels and the data generated as a result are likely sufficiently complete. Then, a case study

is performed using data collected by the Open Signal Maps (OSM) application on real users cell phones in Colorado, so that the limitations of real data collected this way can be understood. Although the data is sparse, geostatistical approaches still perform admirably, suggesting that they may be successfully applied in this domain.

7.1 Drive Test Coverage Mapping

As a first example, consider the data collected from the municipal wireless network in Portland, Oregon. The measurements collected for this network are described in sections 3.1 and 4.2.1. Figure 4.1, shows the collected measurements overlayed on orthoimagery. This dataset is a typical exhaustive drive-test, of the sort commonly performed by network engineers and cell providers in which a vehicle with RF measurement hardware is driven on every publicly accessible street within a 2km by 2km region and RSS values are recorded. This method results in a large number of measurements, contained entirely along public streets. This data is interesting because it is both typical and convenient to obtain, and because it has an obvious source of bias.

7.1.1 Method

Three drive-test data sets are used in this experiment:

- tfa: measurements taken in a suburban environment in Houston, Texas, described in section 3.1.3.2
- pdx: measurements taken in an urban environment in downtown Portland, Oregon, described in section 3.1.3.2
- google: measurements taken in an urban/suburban environment in Mountainview California, described in section 3.1.3.2

These data sets were chosen to provide a large number of representative drive-test measurements with easy comparability to prior work. All data sets were collected passively at 2.4 GHz, using commodity hardware and packet-based measurement strategies. All three data sets have been made publicly available for research purposes.

As a first processing step, the GPS receiver used for measurement reports 6 digits of precision, in order to reduce the data and perform spatial averaging per the method of Lee *et al.* [132], a precision if 5 digits, which corresponds to a resolution of 1.11 meters, or an averaging radius of 8.9 wavelengths at 2.4 GHz is used. For collocated measurements, the median value is used. The next step involves resampling along a uniform equilateral triangular lattice, as described in section 5.4.2. At each point on the sample lattice, the nearest neighbor point is used. Here, two different approaches to sampling are used: careful and aggressive sampling. In careful resampling, if there is a point within 40 wavelengths of the sample point, it is used but if there is not, no measurement is recorded. In aggressive sampling, the nearest point which is no further away than the sampling lag distance is taken. One of the research questions this section seeks to address is how harmful aggressive resampling may be to the fidelity of the ultimate model. These resampled measurements are used to automatically fit a variogram, and Krige a map, using an identical process to that described in chapter 5.

7.1.2 Experiment

The experimental design used here involves resampling, fitting, and mapping measurements from each BS in each dataset, at each of a number of lag distances (h). Lag distances from 10 meters to 500m are used, both with and without careful resampling. Due to the fact that even in careful resampling, measurements may be relocated up to 4 m, 10 m is the smallest reasonable sampling lag. An attempt is made to produce fits for $h \in \{25, 50, 100, 250, 500\}$, although some combinations produce too little data to fit (e.g., $h = 500m$ with careful resampling produces very few measurements in all scenarios), and some produce an intractible number (e.g., $h = 10m$ with will easily exhaust 20 GB of memory). In these scenarios, the experiment proceeds without a fit, and results are derived from those configurations that produce sufficient data and are tractable.

In order to analyze the performance of the fitted models, the method of Robinson *et al.* is used [200]. The original points are used to create an oracle map with a resolution of 0.2 pixels per meter (5 meters per pixel). To determine the fidelity of the fit, each cell on the oracle map (that contains data) is compared with the corresponding cell on the fitted map. Figure 7.1 shows this process schematically. Both the median

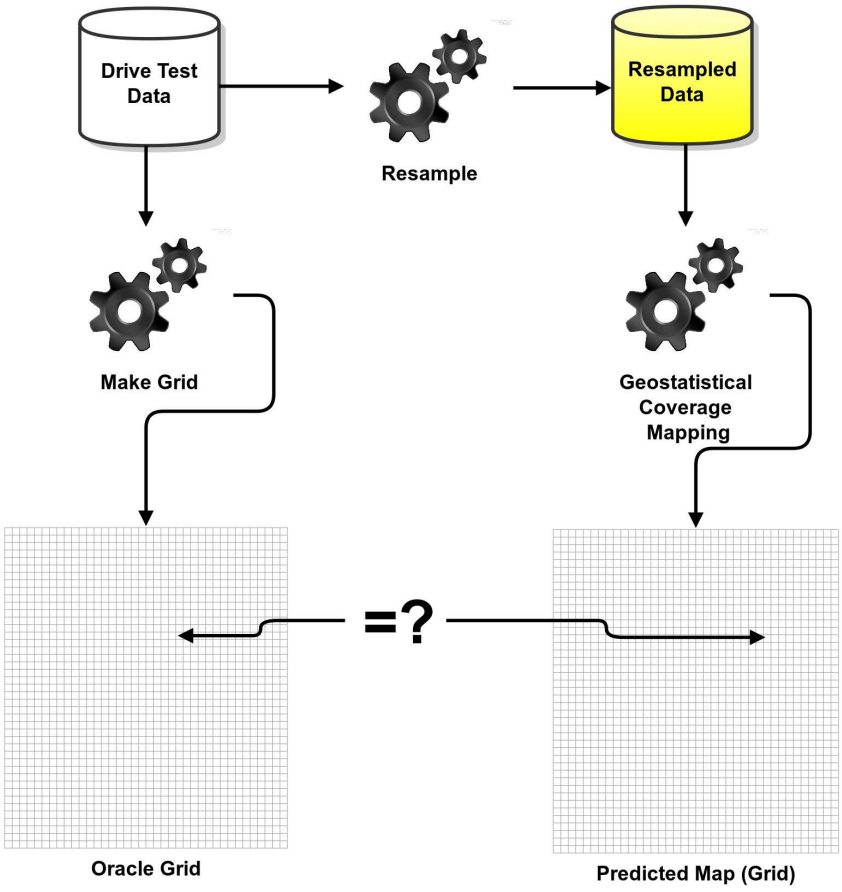


Figure 7.1: Schematic describing validation process for drive-test coverage mapping.

and maximum value are compared, and prediction accuracy for finding holes as well as aggregate RMSE are computed. The same definition of a coverage hole is used here as in [200], measured areas where the SNR dips below 20 dB is are considered to be coverage holes. These performance statistics are calculated in addition to the 10-fold cross validation statistics which measure goodness of fit during the variogram fitting and Kriging process described in chapter 5.

All told, this requires a substantial amount of computation: the complete fitting and kriging process is performed for each combination of BS and lag distance, the corresponding oracle map is extracted from the database of measurements, and the predicted and measured maps are compared. To make this computation feasible in a reasonable amount of time, each combination of BS and lag can be computed in parallel on the Janus supercomputer [156]. Without the level of parallelism offered by this system, performing such an analysis would require many hundreds of days.

7.1.3 Results

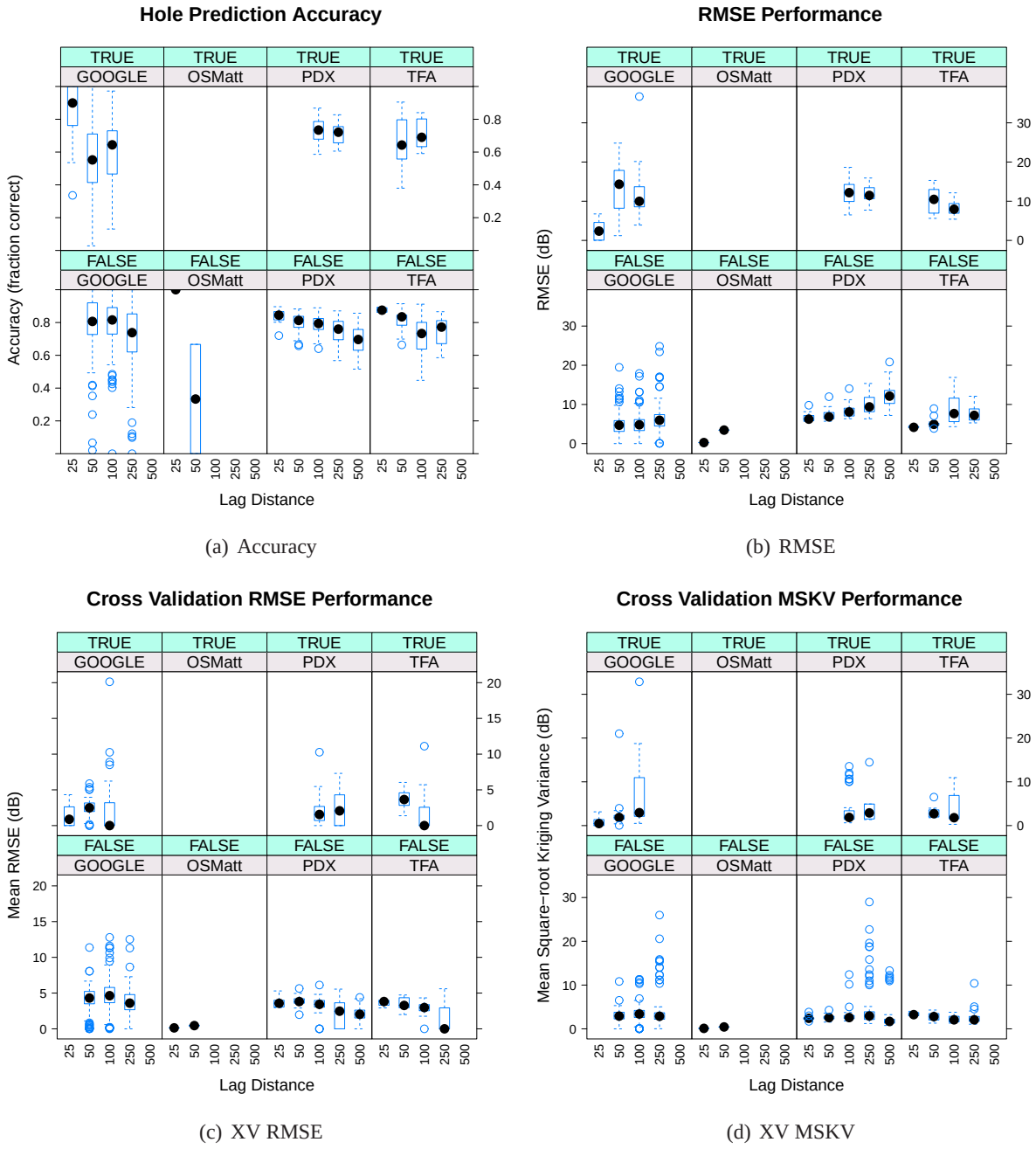
Figure 7.2 shows the overall performance of each resampling for each dataset both with and careful and aggressive resampling. Four metrics are provided, (a) Accuracy, the percentage of holes predicted correctly (areas where the SNR drops below 20 dB), (b) RMSE of the oracle measurements as compared to the predictions, (c) 10-fold cross-validation RMSE from the Kriging process, and (d) 10-fold cross-validation MSKV, also from the Kriging process. The first two metrics describe the resampled map's ability to predict the original data and the latter two metrics describe the residual error in the fitting process.

The first question is whether careful or aggressive resampling produces a more harmonious map. As can be seen in figures 7.2a and 7.2b, in almost every scenario aggressive sampling (bottom row) outperforms careful resampling (top row). The difference is both apparent and has been shown to be statistically significant using a Welch two-sample t-test. This is an unexpected and pleasing result, because it means that it is permissable to "move" measrements by some small amount when resampling without negatively impacting the fidelity of the ultimate map. In fact, the additional data available when doing aggressive sampling appears to produce better predictions.

The next question this section seeks to answer is what is the right resampling density. Clearly this

implies a tradeoff between precision and effort, and the plots support this. For the Google data set, an accuracy of 80% can be acheived with 100m or 50m lags and approximately 75% with 250 meter lags. For The PDX and TFA data sets, close to 90% accuracy can be acheived when resampling at a lag of 25m, and approximately 60% with a lag of 500m. The plots of RMSE tell a similar story, as the sampling lag is increased, the RMSE increases linaerly. Each data set is able to obtain nearly 5 dB RMSE in the best case, and rises to more than 10 in the worst. The cross validation metrics appear to be less sensitive to the sampling lag and regime, however there does appear to be a clear reduction in harmful outliers as the density is increased. A different view of this data is given in figure 7.3, which provides a plot of the Receiver Operating Characteristic (ROC) and Detection Error Tradeoff (DET) space for each sampling lag. Clearly, the choice of lag is application dependent, as one should choose the lag that requires the least amount of work while meeting the accuracy requirements of the application. However, based upon the results here, $h = 100m$ is easy to advocate as a good middle ground between precision and ease of collection for frequencies around 2.4 GHz, supporting the choice of $h = 100m$ in previous chapters.

A final question is how this strategy for selecting samples compares to the state-of-the-art method of iterative heuristic refinement proposed by Robinson [200]. Figure 7.4 addresses this, by explicitly showing the relationship between sampling density and hole prediction accuracy. As compared to the method of Robinson almost exactly the same performance is acheived with a similar number of samples. For the Google dataset, Robinson's framework is able to acheive approximately 75% accuracy with slightly more than 10 measurements per km^2 and nearly 80% accuracy with 70 measurements per km^2. The same result is acheived here, and indeed the performance appears to flatten around 0.8 with increasing samples providing no improvement in performance (incidentally, the 65 measurements per km^2 result corresponds to the $h = 100m$ resampling). With the TFA dataset, Robinson's framework acheives slightly better than 70% accuracy with 15 measurements per km^2 and is able to ultimately acheive slightly better than 80% with approximately 60 measurements. The method proposed here is acheives a higher starting accuracy, but appears to acheive the 80% mark more slowly. Although the PDX dataset was not studied by Robinson, the results are similar, and perhaps more consistent than the others; an average accuracy of 80% is acheived with approximately 70 measurements and tops out near 85% with more then 600 measurements. Overall these results, and their



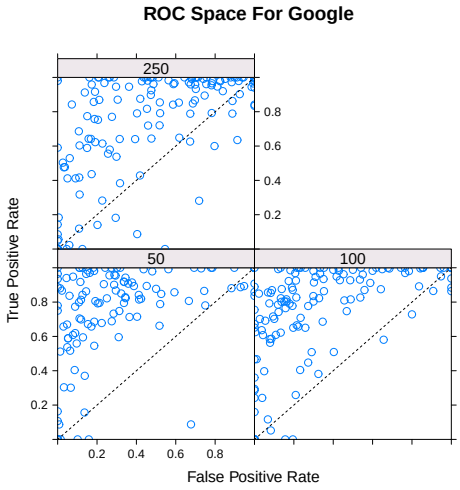
(a) Accuracy

(b) RMSE

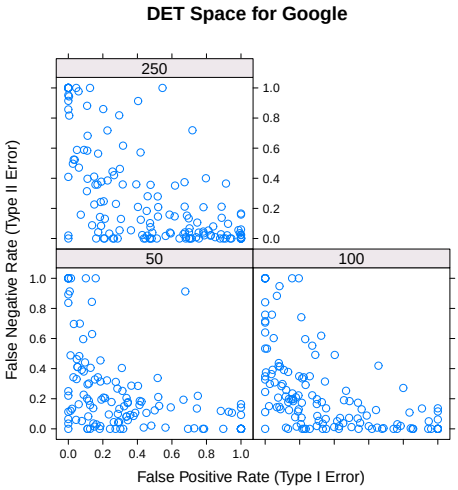
(c) XV RMSE

(d) XV MSKV

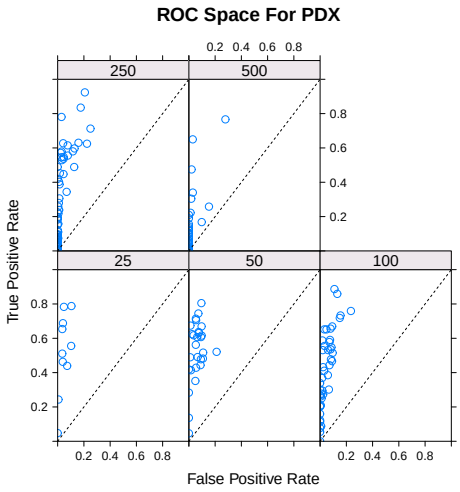
Figure 7.2: Performance results for different resampling lags and careful and aggressive resampling using four metrics of interest.



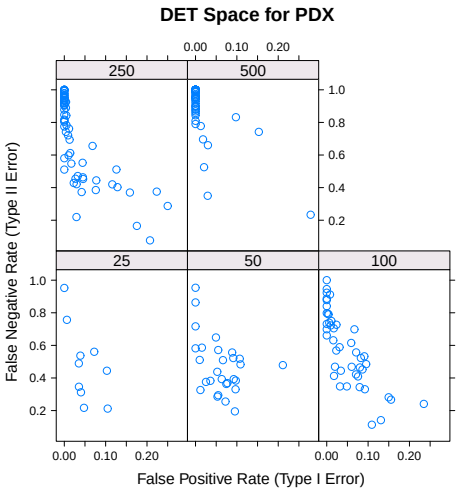
(a) Google, ROC



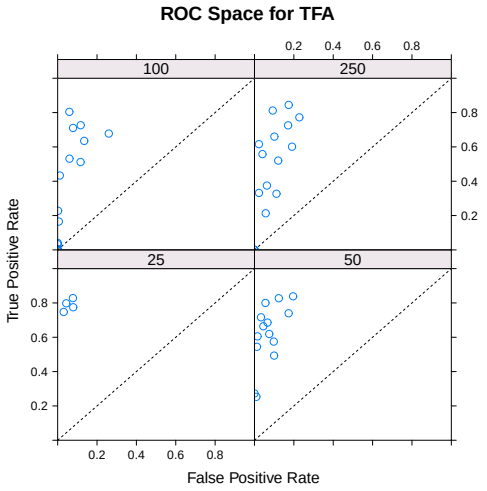
(b) Google, DET



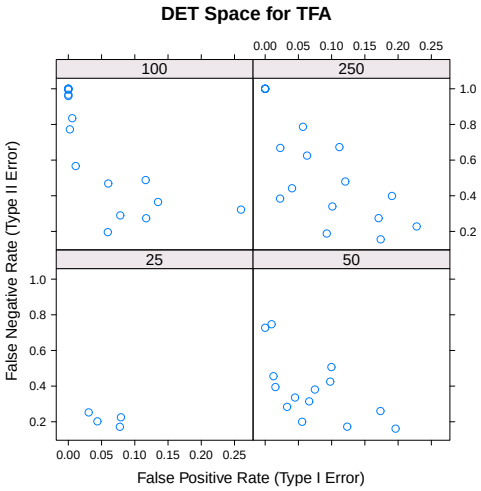
(c) PDX, ROC



(d) PDX, DET

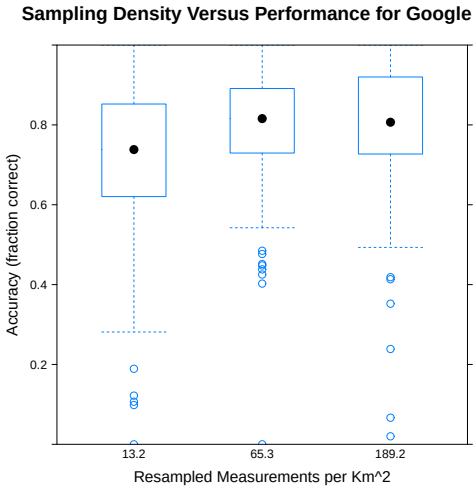


(e) PDX, ROC

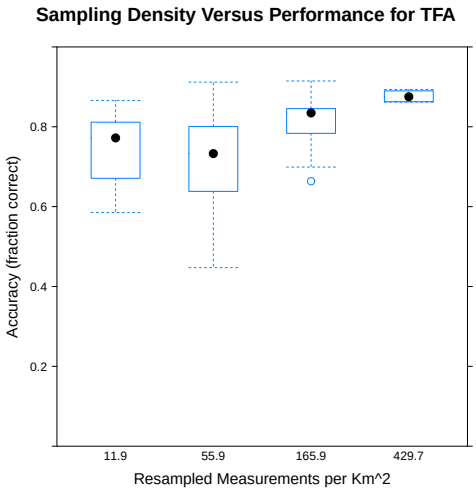


(f) PDX, DET

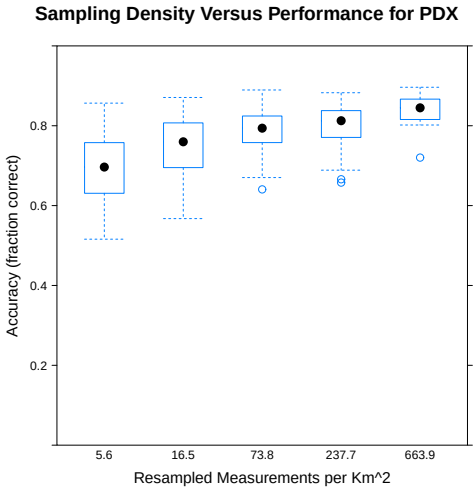
Figure 7.3: ROC and DET curves for each dataset using “aggressive” resampling.



(a) Google



(b) TFA



(c) PDX

Figure 7.4: Hole prediction accuracy as a function of sampling density.

strong agreement with the performance of Robinson's method, hints at an upper-bound in terms of accuracy using this metric. However, the most important conclusion these results highlight is that approximately the same performance can be acheived with Kriging and iterative heuristic refinement. Since Kriging provides a substantially richer perspective on propagation (providing a value distribution at each point, and not simply a boolean covered/not-covered value), it seems easy to advocate for this type of application in almost every circumstance.

7.1.4 Discussion

This section described and analyzed the application of the geostatistical methods proposed in chapter 5 to the problem of mapping coverage using drive-test data as input. This is an important extension becuse drive test measurements are generally easy to obtain and are widely used by cell providers for site-surveying. In order to cope with the intrinsic sampling bias of making measurements in city streets, this section proposed a resampling methodology where the closest measurement is taken at each vertex on a uniform equalateral triangular sample. Ultimately it was found that there is a linear relationship between the resampling density used and the predictive ability of the resulting map. The mean error observed here is typically larger than was observed in the case studies described in section 5, however those studies involved the careful placement of samples and measurement with calibrated equipment. One higher-order result of this section may be that while *yout get what you pay for* in terms of measurements, even rough measurements can produce a reasonably accurate map when an appropriate resampling scheme and statistical modeling process is used. Indeed, the error is substantially smaller than would be acheived with an *a priori* predictive method or simple data fitting approach (see chapter 3), while producing nearly identical results to the iterative heuristic refinement method proposed in [200]. Figures 7.5, 7.6, and 7.7 show the map-combined maps for these datasets, which bring to light the rich detail that can be derived of the coverage using these measurements, especially as compared to the simplistic maps produced by *a priori* models (e.g., see figure 3.28) and iterative heuristic refinement (e.g., [200]). The next section will take this idea a step further by introducing and investigating the idea of crowd-sourced coverage mapping, where many individuals contribute to the data collection process voluntarily.

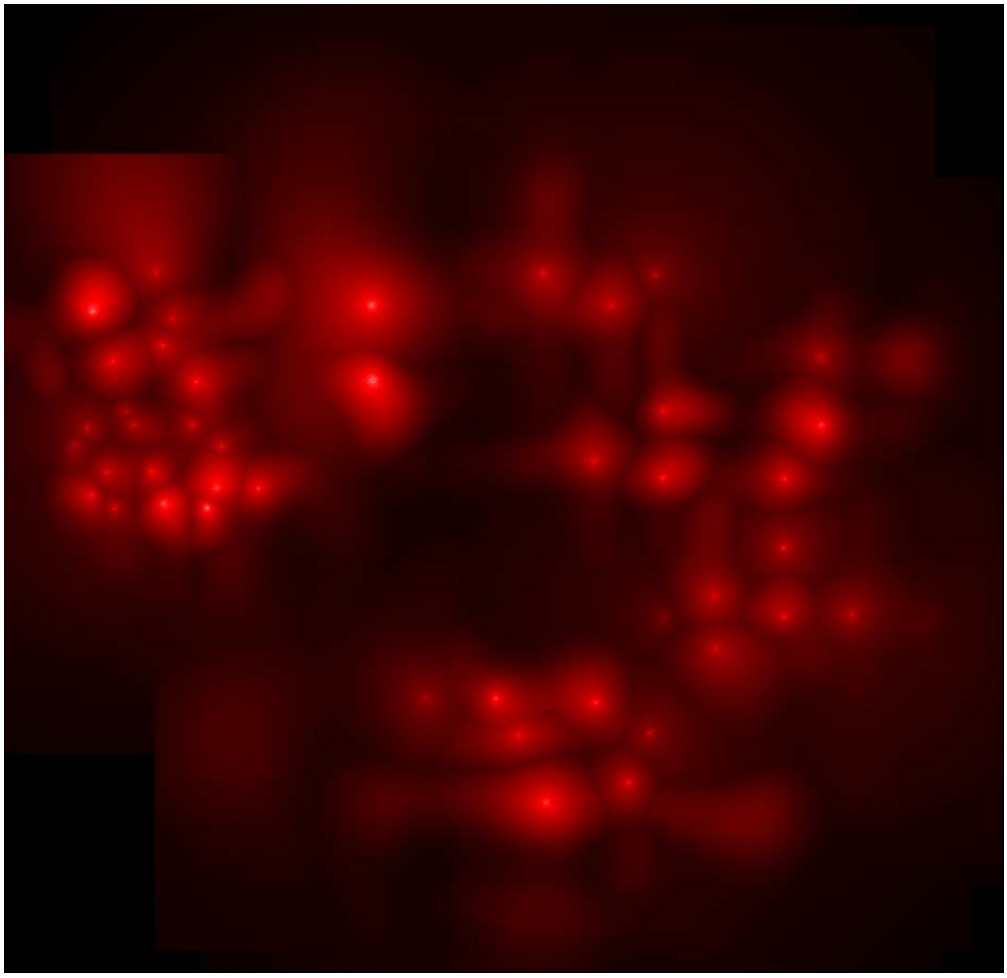


Figure 7.5: Map-combined (maximum combining) Kriged coverage map for PDX Data using best variogram and h=100m.

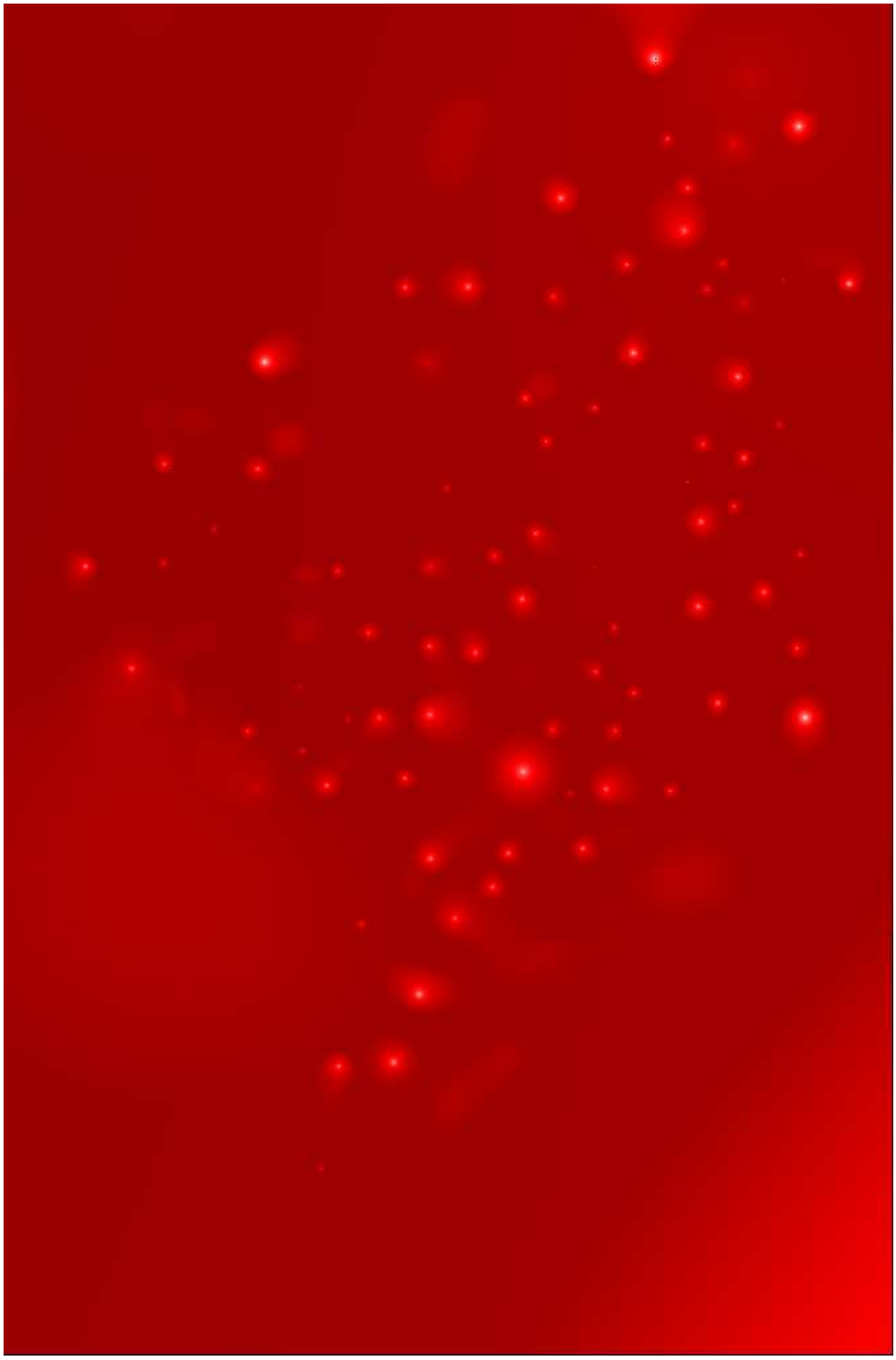


Figure 7.6: Map-combined (maximum combining) Kriged coverage map for Google Data using best variogram and h=100m.

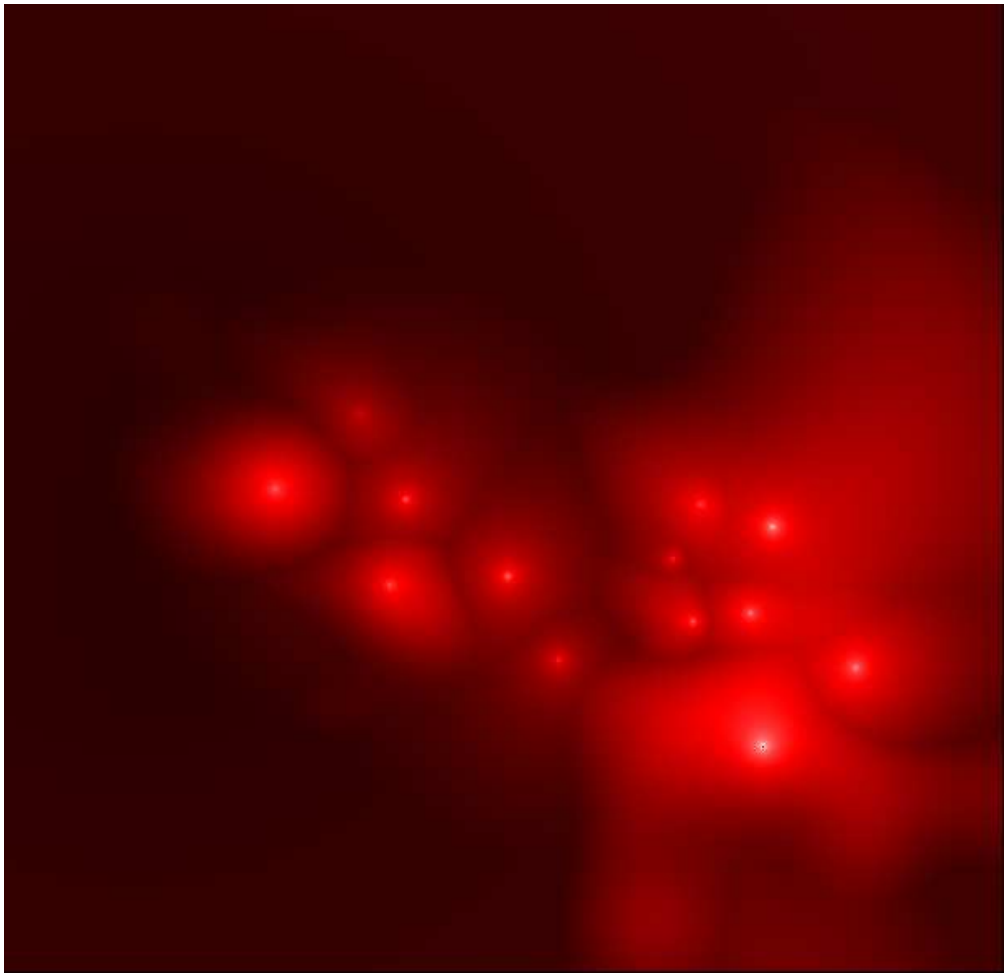


Figure 7.7: Map-combined (maximum combining) Kriged coverage map for TFA Data using best variogram and h=100m.

7.2 Crowd-Sourced Coverage Mapping

This section introduces the topic of crowd-sourced coverage mapping, where mobile devices and their users are solicited (wittingly or unwittingly) to collect measurements of a wireless netowork. The idea of using mobile devices for sensing applications is not new, and in fact has been applied to a large number of interesting applications. For instance, in [70] the authors use sensing mobile devices to generate better cycling maps, and in [137] the authors describe a system for mapping sources of noise in an urban environment to enable the discovery of quiet places. [126] provides a survey of various mobile sensing applications. The prospect of using these mobile devices to map the radio environment is a fairly recent proposal which has seen some modest interest (e.g., [244, 47]). This section will try to understand the feasibility of crowd-sourced coverage mapping in the context of the geostatistical approach taken in this thesis. In the next section, the key question of how many participants would be needed to map an area of typical size and complexity is analyzed via simulation using a human mobility model. Then in section 7.2.2, real crowd-sourced traces collected by the OSM application will be analyzed to understand the practical difficulties of fitting sparse and incomplete data using geostatistical methods.

7.2.1 Feasibility Study using Simulated Mobility

In order to understand how a crowd-sourced sampling scheme might scale to a large region with a large number of users, this section looks to human mobility models and verification via simulation. For this, the Self-similar Least Action Walk (SLAW) model proposed by Lee *et al.* is used [130]. This model is based on an extensive modeling and data-fitting campaign using GPS measurements from theme parks, college campuses and cities. SLAW assumes that there are some fixed number of way-points, which the individuals being simulated visit in a stochastic manner. For trace generation, at each time step, some number of users will move from one waypoint to another along a straight line path and reside there for an amount of time also determined by the model.

To simulate the dynamics of the CU campus, 6,205 unique users were modeled. This constitutes 20% of the students and faculty at the University, and is meant to approximate a rough lower bound on the

number of individuals present on campus on a given day. The main CU campus has 145 buildings and other facilities, which are used as way-points. The campus is roughly a right triangle, with a long side at 1.5km and s short side of 850m. Hence, for this simulation, a square space 1.5 km on a side is simulated, which is approximately twice the size of the existing campus. This further skews results towards a conservative estimate. It is assumed that a building or facility has a radius of 20m (i.e., any movement within 20m of a waypoint is located at that waypoint), and that users pause at waypoints for some amount of time between 1 minute and 2 hours (a typical class length). The SLAW model also requires some additional model-specific parameters that control the stochastic self-similar nature of the generated walks. For these, values that are similar to those observed in data collected at the University of California San Diego (UCSD) and Korea Advanced Institute of Science and Technology (KAIST) are used: $\alpha = 1.6$ and $\beta = 0.8$. To gain enough data, MATLAB code provided by [130] is used to generate a 170 hour trace. The first 50 hours of the trace are excluded to avoid transient behavior and the remaining 120 hours (5 days) are used below.

In order to understand how the number of users and the time elapsed effects the sampling coverage, the number of users participating and the duration of the study are varied. A random subsample of users from 0.01% to 100% is used to generate a trace with a max time from one hour to 150 hours. For each trace the study area is subdivided into a 5 m grid (approximately 40 wavelengths at 2.5 GHz). How many times the path of the particular user enters a given grid square is counted. Perfect coverage for a given participation rate and time would have every square entered at least once. In an imperfect sample, those areas not visited by a user form measurement “holes” and limit the final resolution (and prediction accuracy) of a map generated from the sample coordinates.

Figure 7.8 gives an example of the aggregate coverage with several different participation percentages and durations. The left hand plots are colored so that grid cells which see the largest number of visits are darker (number of visits is on a log scale). The right hand plots give a boolean version of this map, where cells that have seen any visits are colored purple and cells that have not seen any are cyan. With just one user and one hour, hardly any of the map is covered. However, with a modest increase in the percentage participating to 6% (372 individuals) and a single work day at 7 hours, much of the map is covered. After 81 hours, with 96% of people participating (5957 individuals), the sample is completely covered except near

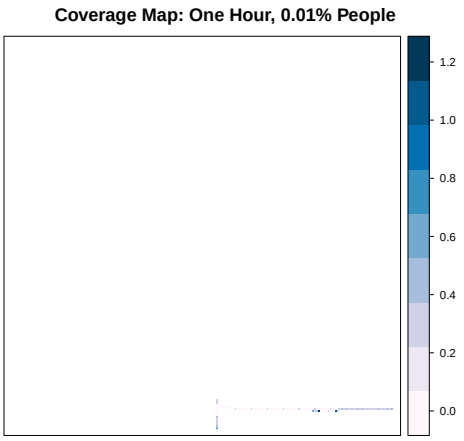
the edges.

Figure 7.9a shows the number of holes remaining as a function of the fraction of participating users and time elapsed. This plot is on a logscale, since there are a very large number of holes when there are few participants or a small amount of time. To find holes,the sample grid is searched for contiguous regions that have zero measurements. By increasing the percentage of participating users, or the duration of the study, the number of holes decreases exponentially. Figure 7.9b plots a different metric, which is the maximum "effective hole diameter" as a function of participation and time. To determine the effective diameter of a hole, we calculate the width of the hole and the height of the hole. The effective diameter is the diagonal line scaled by the fraction of cells that are empty in the square bounded by the height and the width:

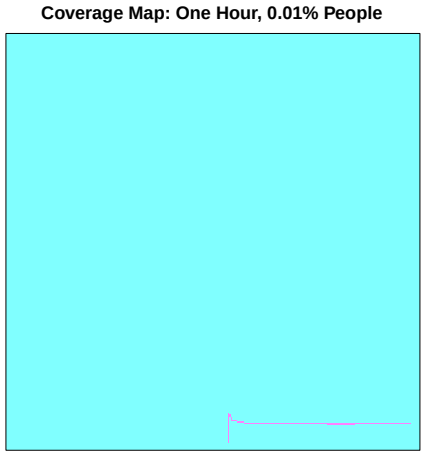
$$d = (a/n) * \sqrt{h^2 + w^2} \tag{7.1}$$

where a is the number of cells that are empty in the square region contained in a box centered on this hole of height h and width w and n is the number of cells total in that square (i.e., $n = h * w$). Although this is not a perfect description of the shape of a given hole, it acts as a conservative single value description of the limitations of a given sample. Compared to figure 7.9a, we can see that many levels of participation and durations that produce a large *number* of holes, do not produce holes of a very large size. In fact, most experiments except the shortest or those with the least participation, have a maximum effective hole diameter of 50*5 = 250 meters. Finally, figure 7.9b plots the number of holes greater in diameter than 50 squares as a function of participation and duration.The majority of cases have one or two "large" holes, which are presumable at the edges as in figure 7.8e and 7.8f.

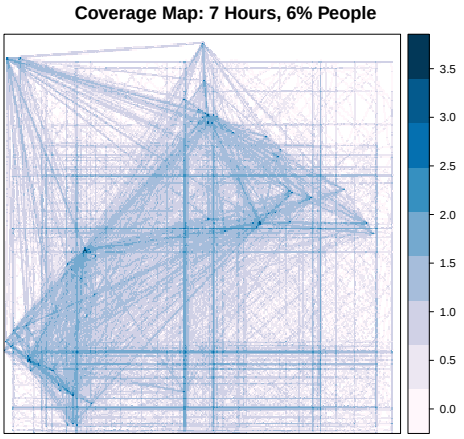
From these experiments it can be concluded that a reasonably small fraction of the CU students and faculty would be required to participate in a crowd-sourced mapping campaign in order to collect sufficient samples required to produce a high resolution coverage map. For instance, with a total of 14.2% of the campus participating (4506 individuals), the complete campus could be mapped within three hours. And, with 4.2% total participation (1304 individuals), a similar level of coverage would be acheived after 25 hours. Although promising, there are few criticisms that can be made of these results which may affect their



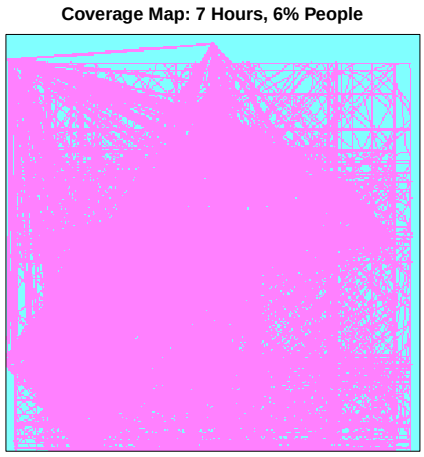
(a) 0.01 Percent, 1 Hour



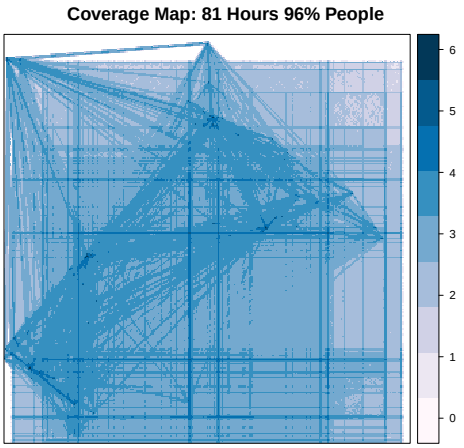
(b) 0.01 Percent, 1 Hour



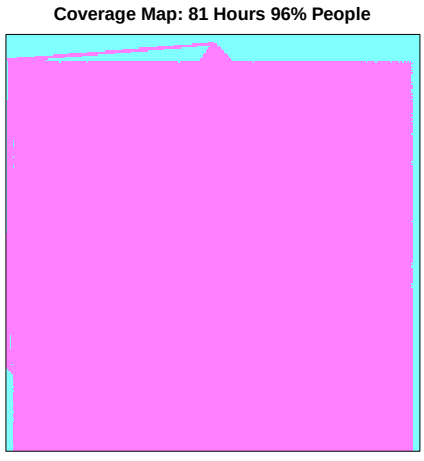
(c) 6 Percent, 7 Hours



(d) 6 Percent, 7 Hours

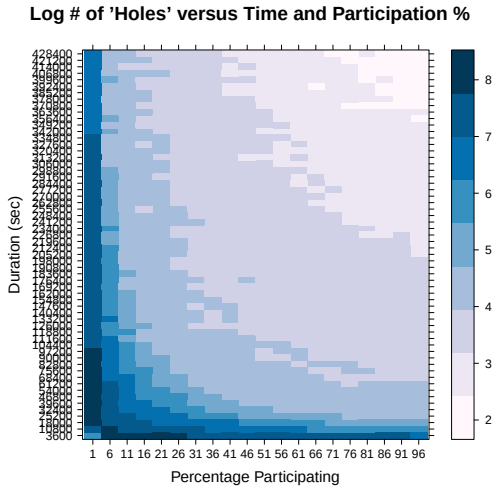


(e) 96 Percent, 81 Hours

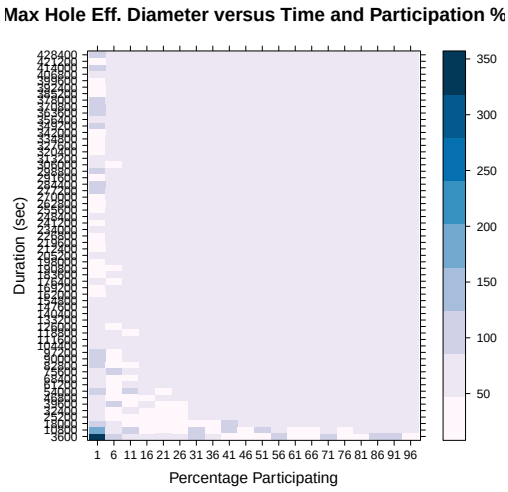


(f) 96 Percent, 81 Hours

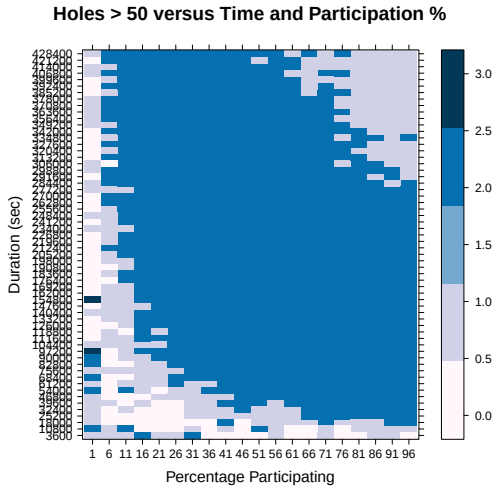
Figure 7.8: Sample coverage as a function of percentage of participating users and length of experiment. The left-hand plots show the number of visits to each 5 meter cell on a log scale. The right-hand plots show a boolean map of cell coverage.



(a) Number of Holes



(b) Max Hole Size



(c) Number of Big Holes

Figure 7.9: Sample coverage as a function of percentage of participating users and length of experiment.

generalizeability:

- Obstructions are not modeled. In this simulation it is assumed that users can walk directly between waypoints on a straight line without routing around obstructions, people, and other waypoints. In realitiy, users likely constrian themselves to predefined paths and do not visit some (i.e., fenced and walled) areas. This would cause more holes to exist, but centered at the points where information is likely least valued (i.e., users do not go there, so a high resolution map of that particular spot may not be valuable).

- The CU campus is very user-dense, with more then 30,000 individuals sharing the same 2.25 square kilometers. Areas with fewer users and sparser waypoints would be sampled less readily. However, the crowd-sourcing model is inherently democratic: the areas with the most users will recieve the most samples and therefore have the most accurate measurements. In scenarios where measurements are being used to plan future deployments, a crowd-sourced data collection may need to be augmented with directed measurement in the area of interest.

- It is assumed that every visit to a given grid cell produces a usable measurement, when in practice mobile devices may produce noisy measurements, or fail to make useful measurements in some location (e.g., a failure to obtain good location information through GPS constitutes a general failure of measurement. However, we argue that there is a steady trend of improvement in the measurement and locationing abilities of mobile devices and any limitations along these lines are likely to be mitigated in the near future by the advent of better mobile systems.

Overall, the results in this section make a compelling argument in favor of the abilities of crowd-sourced coverage mapping systems. Given a reasonably small fraction of participating users (4 to 15%) a fairly large and diverse area can be extensively measured in a reasonable amount of time (three to 25 hours). One can imagine such a system producing real-time wireless coverage maps (perhaps using a geostatistical interpolation method similar to that described in chapter 5 that regenerate periodically in response to measurements provided by users in real time. The next section looks at maps generated from real measurements collected by a prototype crowd-sourced sampling system.

7.2.2 Case Study: Open Signal Maps

In order to understand how well the crowd-sourced coverage mapping method may work in practice, this section describes experiments using real data collected by the OSM application [101, 47]. The OSM application is a project of the United Kingdom (UK)-based web development firm, Staircase3 [212]. Their application runs on Google Android-based phones. Users voluntarily download and install the application which collects measurements of signal strength and location in the background and sends them to the company via a web-service. Figure 7.10 gives an example of a map generated from crowd-sourced measurements on the OSM website. The mapping method used is basic heatmapping without interpolation. The heatmapping regime assumes that an absense of measurements indicates weaker signal and places red areas (strong signal) over regions with many strong meausurements.

In October 2011, a meeting was held with developers at Staircase3 to discuss the possibility of a research data collaboration and in particular the prospect of using geostatistical techniques to map the data. The developers agreed to share all the data they had collected contained within the state of Colorado for this analysis. The data contains measurements of multiple networks by a number of distinct users. For the purpose of this case study, results for the AT&T network are presented, as it was the most prevalent network in the measurements, and a subset of measurements confined to the city of Boulder, within the bounding polygon defined by the points $(39.995057, -105.249195, 40.011658, -105.277476)$ were used. Because the data is sparse, and often confined to straight lines (similar to measurements made in drive-test studies), "aggressive" resampling was used.

The complete dataset contains measurements from 581 distinct BSs with as few as one and as many as 1,257 measurements each. These measurements were collected by 190 distinct devices, and 74 total devices making measurements of the AT&T network. Within the bounding polygon defined above, 50 unique AT&T BSs are observed to varying degrees by 13 unique devices. Figure 7.11 shows the resampled measurements at different lags for a representative BS. Clearly these measurements are data-sparse, especially considering that this BS has more measurements present in the dataset than most. Figures 7.12 and 7.13 show the excess map and coverage map for this BS, mapped using the same method described in section 5. Figure 7.19 shows

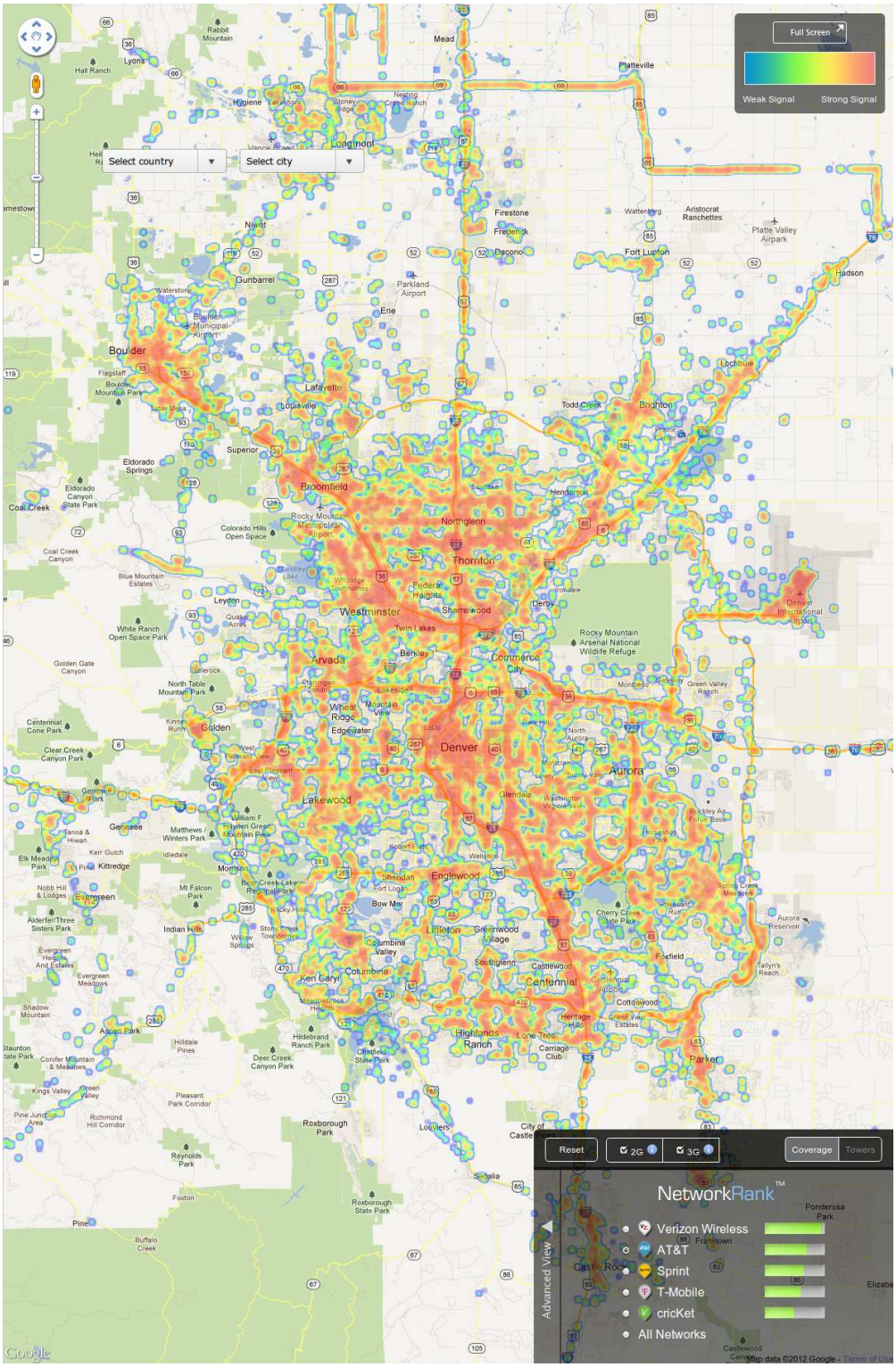


Figure 7.10: Open Signal Maps measurements for AT&T network overlayed on Google orthoimagry.

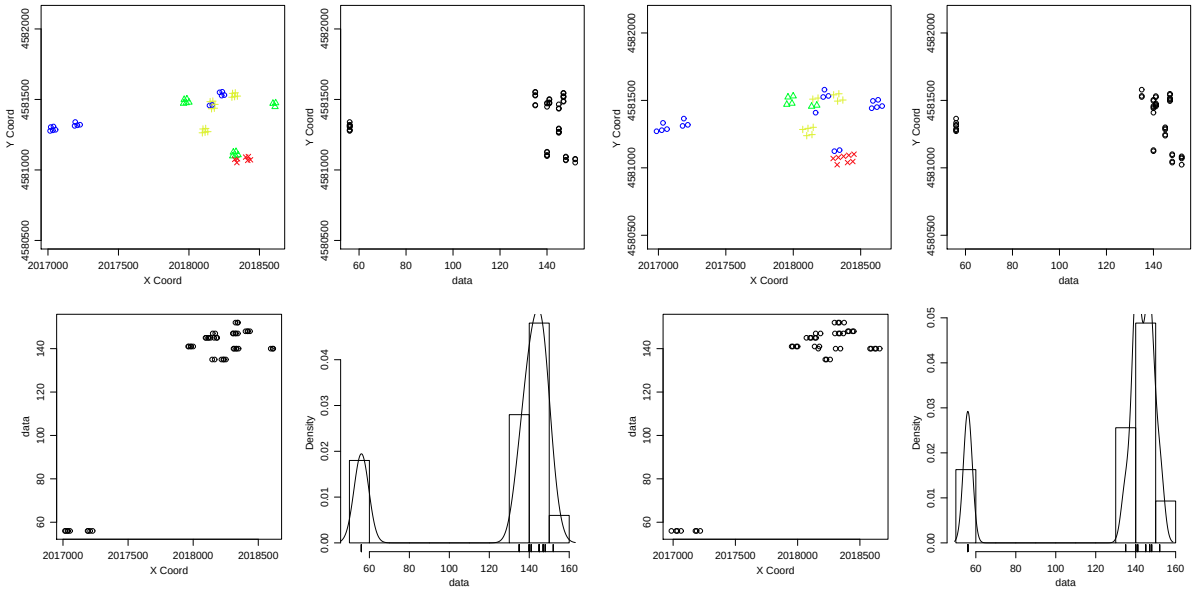
Lag (m)	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
25	cubic	3290.48	25.00	6259.02	417	FALSE/FALSE	5.54	4.02	8.79
50	cubic	2182.09	75.54	190.31	361	FALSE/TRUE	8.93	9.24	4.78
100	cubic	29648.22	41.28	49124.86	219	TRUE/TRUE	6.59	10.74	8.22
250	gaussian	792.46	237.22	606.03	83	TRUE/TRUE	17.09	14.74	5.95

Table 7.1: Best fit statistics for variogram fitting of resampled OSM AT&T data.

the threshold-based map using an SNR of 20 dB as the coverage criterion. These maps highlight the fact that each resampling density produces a considerably different map when the data is sparse. Table 7.1 proves cross-validation performance results, which show that the smaller lag distances (i.e., denser resampling) generally produce less residual error, with a RMSE of 4.02 dB acheived with $h = 25$, and gain of 8.79 dB over simple log/log data fitting.

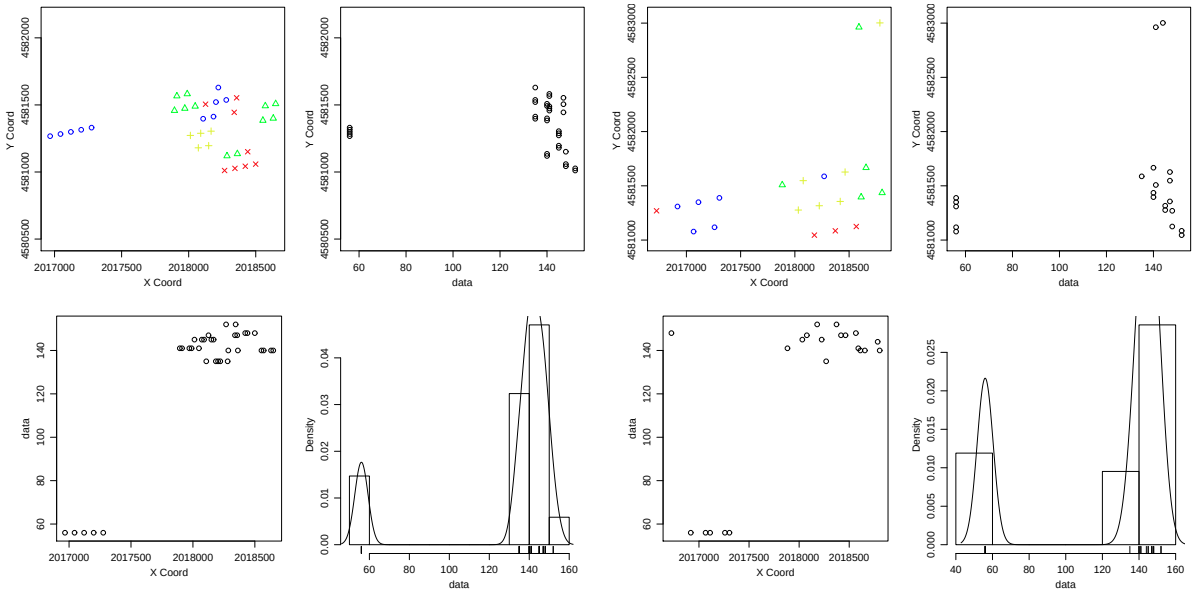
The other 49 BSs have similar fit statistics, although those with very little data (i.e., only a handful of actual observations) fit quite poorly as would be expected. Figures 7.15 and 7.16 show the map-combined maps for all the BSs combined. These maps make clear the fact that most measurements present are from a small fraction of the 50 BSs, presumbaly because those users participating in the measurement spend most of their time in the coverage of those cells. In this way, this dataset is similar to the LTE case study in chapter 6, where the combined maps provide greater insight into the coverage of the network since individual cells only provide a partial picture (and many cells have been insufficiently measured). However, due the limited number of measurements for most BSs in this map, the picture of coverage is likely far from complete—more participation is needed in order to obtain a sampling coverage sufficient to produce a consistent and meaningful map. Simply put, the 13 participating users are not quite 1% of the 1304 participants shown in prior section to be the number of individual participants required to sufficiently map a slightly smaller area within 25 hours.

Figure 7.17 provies an aggregrate view of performance for all 49 BSs, which is similar to the analysis done in the previous section. This plot shows that in spite of the data sparsity, the predictive ability of the maps is still fairly strong, at least with respect to the measurements that are available. The mean coverage hole prediction accuracy is approximately 90%, and mean RMSE is around 7 dB. The residual error in the fits is small, generally less than 5 dB. The ROC and DET plots are given in figure 7.18, which show similarly



(a) h = 25

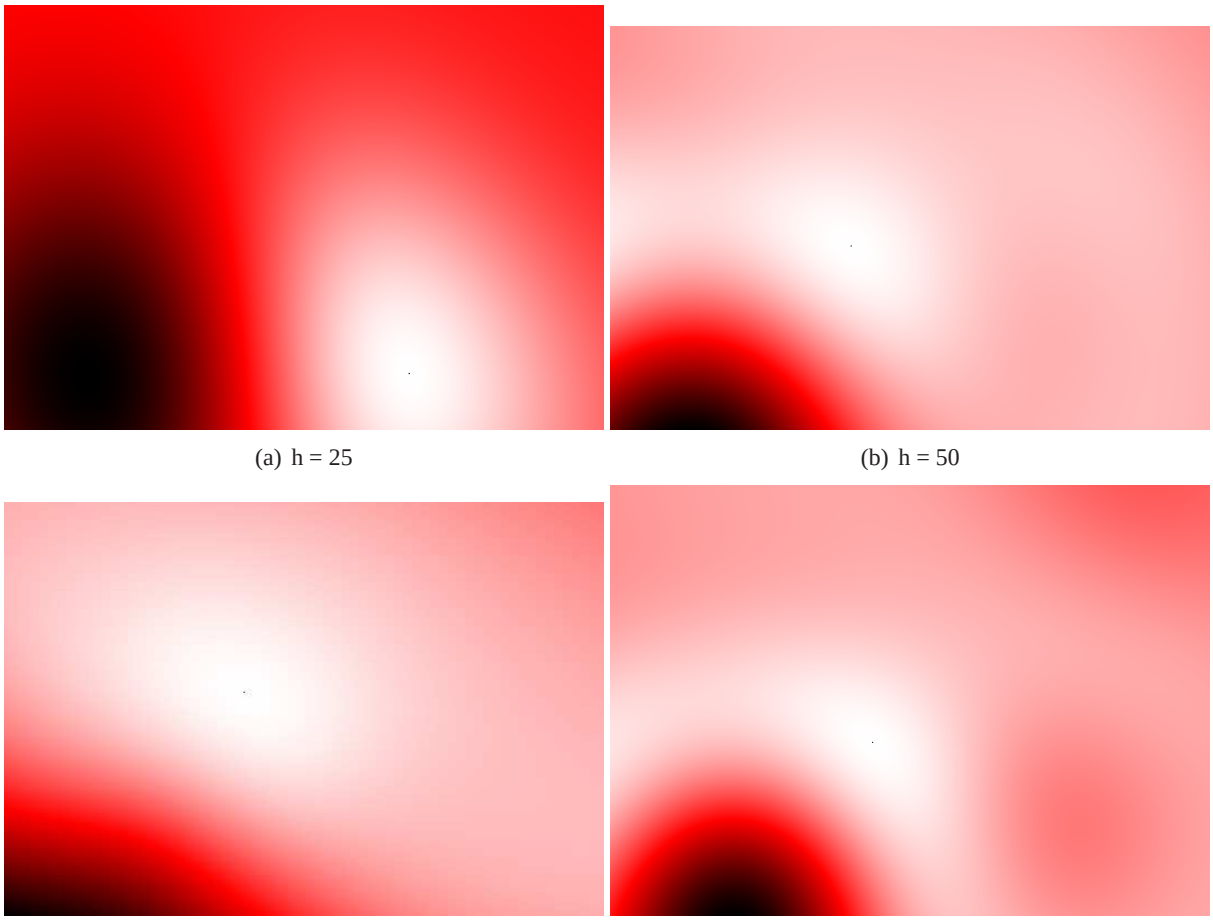
(b) h = 50



(c) h = 100

(d) h = 250

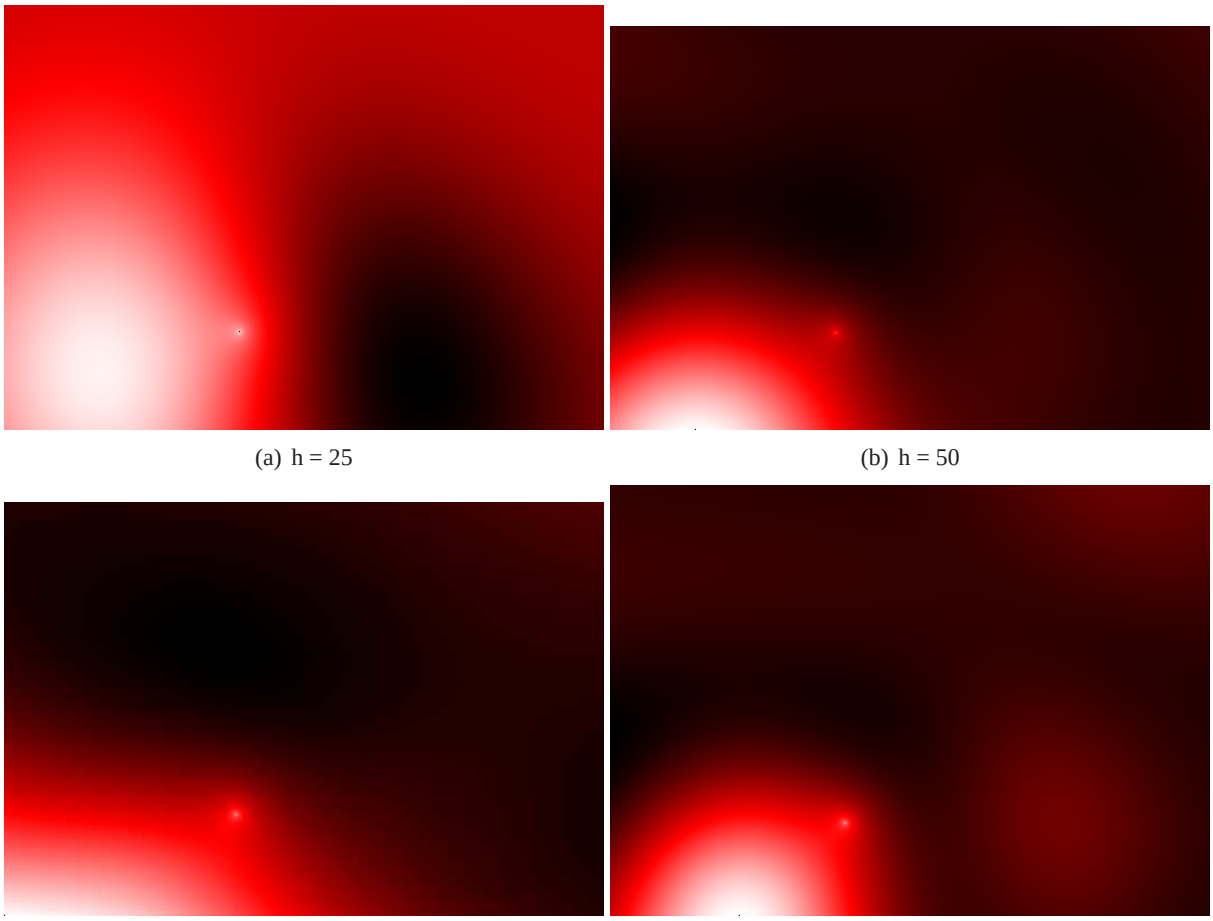
Figure 7.11: Performance results for OSM data at different resampling lags using four metrics of interest.



(a) h = 25 (b) h = 50

(c) h = 100 (d) h = 250

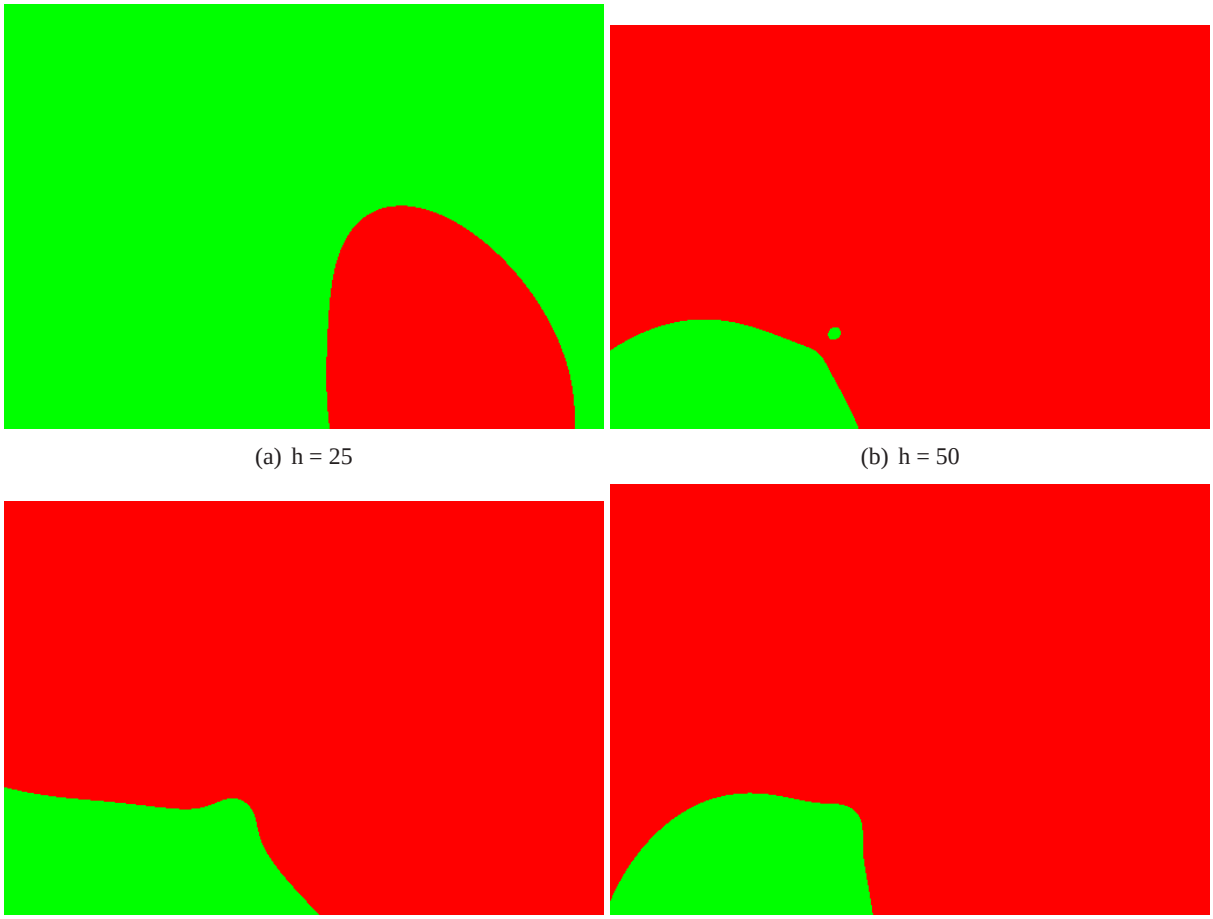
Figure 7.12: Performance results for OSM data at different resampling lags using four metrics of interest.



(a) h = 25 (b) h = 50

(c) h = 100 (d) h = 250

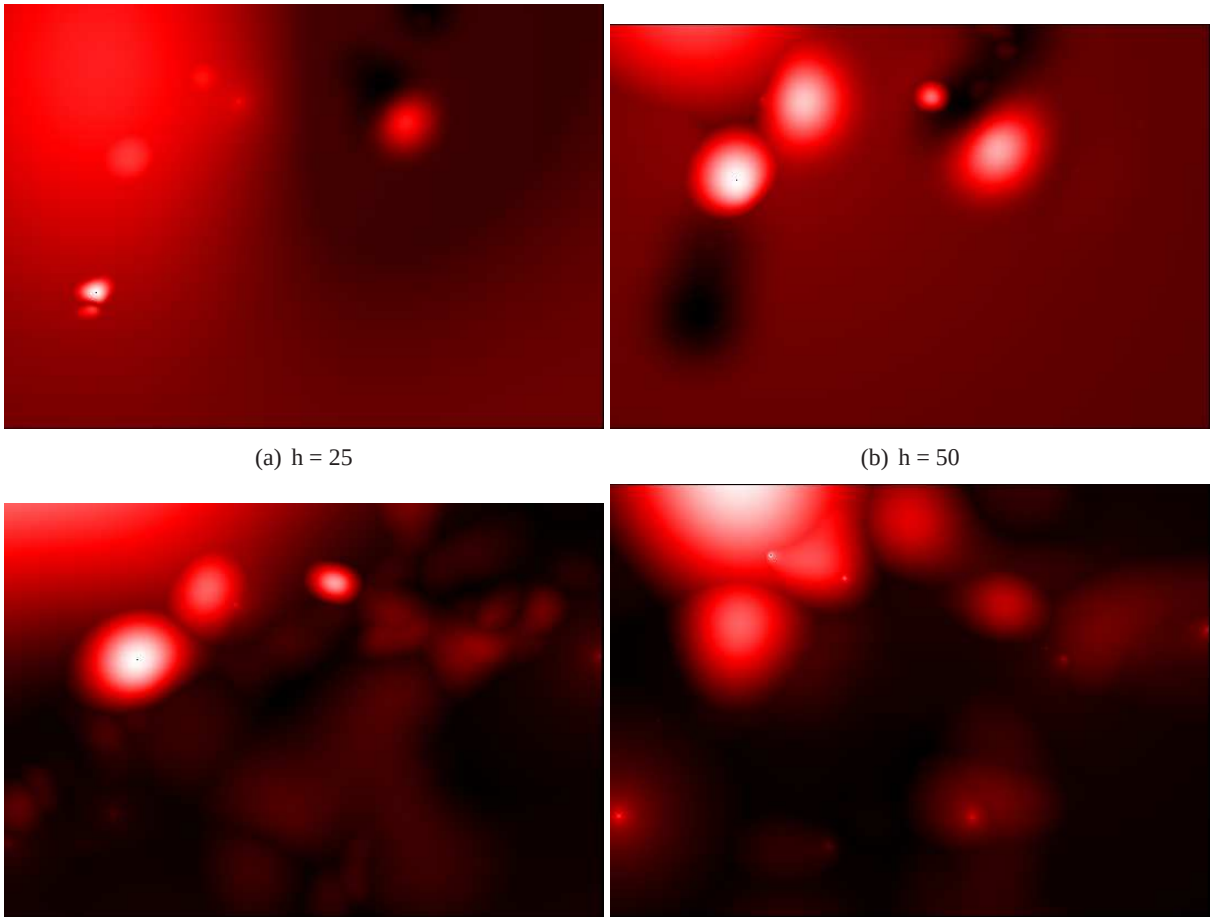
Figure 7.13: Performance results for OSM data at different resampling lags using four metrics of interest.



(a) h = 25 (b) h = 50

(c) h = 100 (d) h = 250

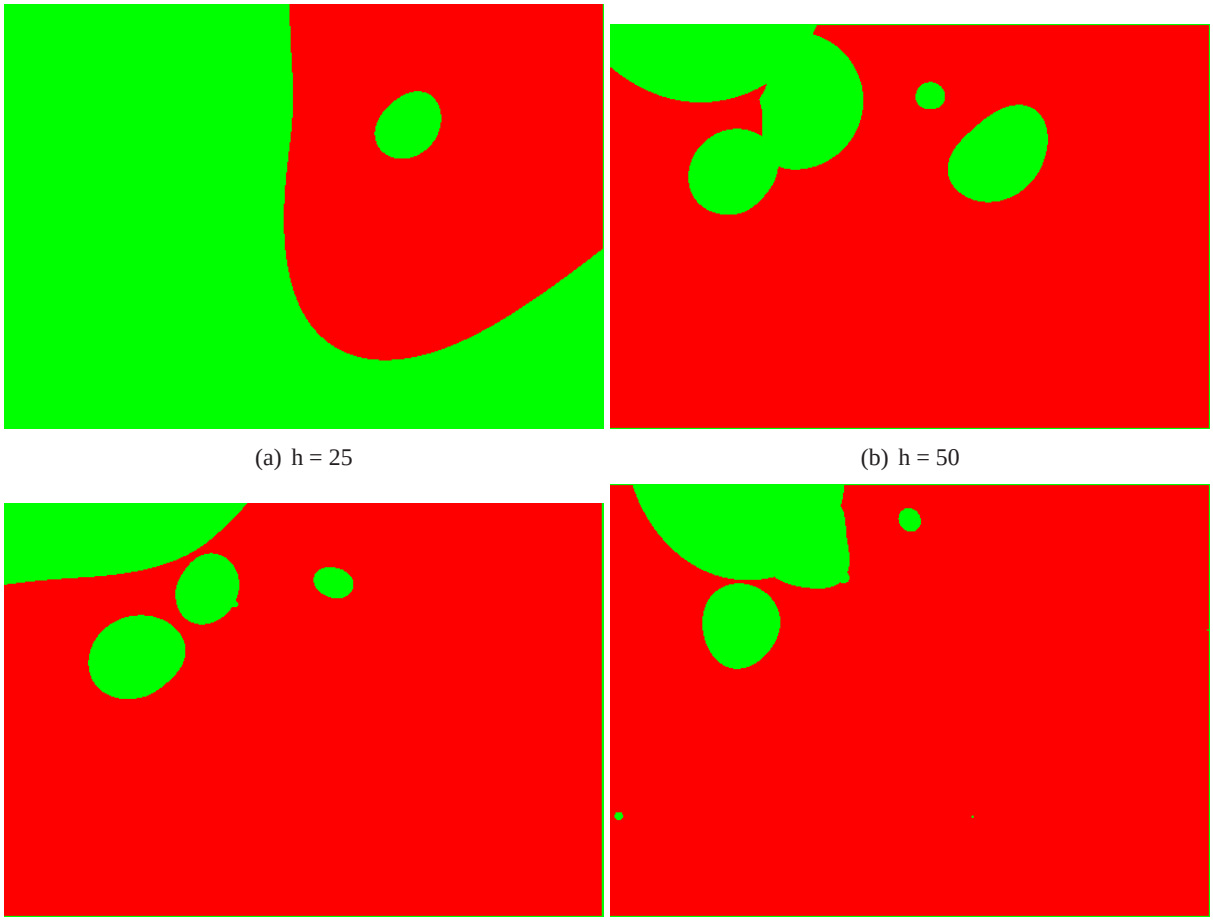
Figure 7.14: Performance results for OSM data at different resampling lags using four metrics of interest.



(a) h = 25 (b) h = 50

(c) h = 100 (d) h = 250

Figure 7.15: Map-combined maps using maximum-based combining for OSM data.



(a) h = 25 (b) h = 50

(c) h = 100 (d) h = 250

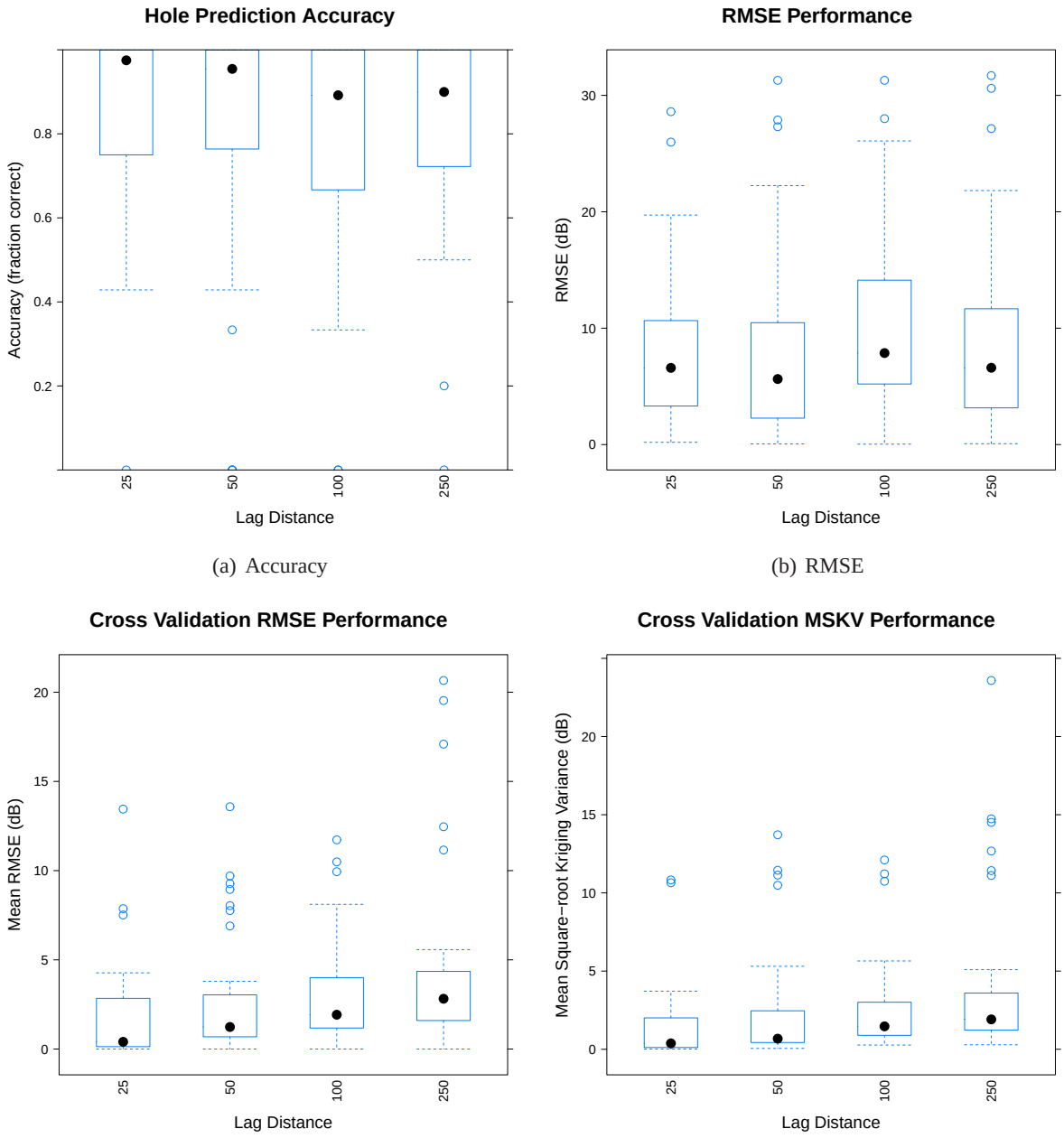
Figure 7.16: Map-combined threshold maps using maximum-based combining for OSM data.

strong results in terms of hole prediction, with very few false negatives and a reasonably small number of false positives. Finally, figure 7.19 shows the relationship between resampling density (samples per km^2) and hole prediction accuracy. Here the number of samples to acheive an accuracy of nearly 90% is less than 60 per km^2, which is quite good by the standard set for municipal WiFi networks in [200].

Overall, these results bring to light two conclusions. First and foremost, geostatistical coverage mapping appears to be a reasonable way to produce maps of crowd sourced measurements. However, real crowd-sourced data is extremely sparse as compared to data collected for the express purpose of coverage mapping. A larger fraction of users need to participate in order to acheive the sampling density required to generate a coverage map that is as detailed as would be necessary for it to be useful in practice. Although the previous section showed that only a small fraction of all people need to participate to create a complete map, that fraction does not appear to be met in this dataset. With the current level of measurement, only broad conclusions can be drawn (e.g., which BSs are most prevalent in a given area and their general propagation pattern). For the cell providers themselves, extensive measurements could be collected simply by requiring users to provide periodic measurements of the channel to a central source. However, such an application may have privacy implications, exposing the location and movement behavior of end-users to the providers. In addition to increased participation, further work is needed to understand the practical fidelity of mobile phones as measurement devices. However, once the data is available, and at sufficient density, crowd-sourced coverage mapping using geostatistical approaches appears to be a feasible design.

7.3 Summary and Conclusion

This chapter took a practical look at geostatistical coverage mapping by considering two methods of measurement that prioritize ease of collection and convenience. Although a careful and principled sample is clearly the best strategy for producing an accurate coverage map, the practical adoption of the methods proposed in this thesis requires that they work with existing measurement methodologies as well. First, drive test measurements were studied as a basis for mapping. It was found that resampling measurements on an equilateral triangular lattice is a reasonable way to cope with the inherent sampling bias in these datasets. The resulting Kriged maps provide a rich picture of the RF landscape, with a small error as compared to



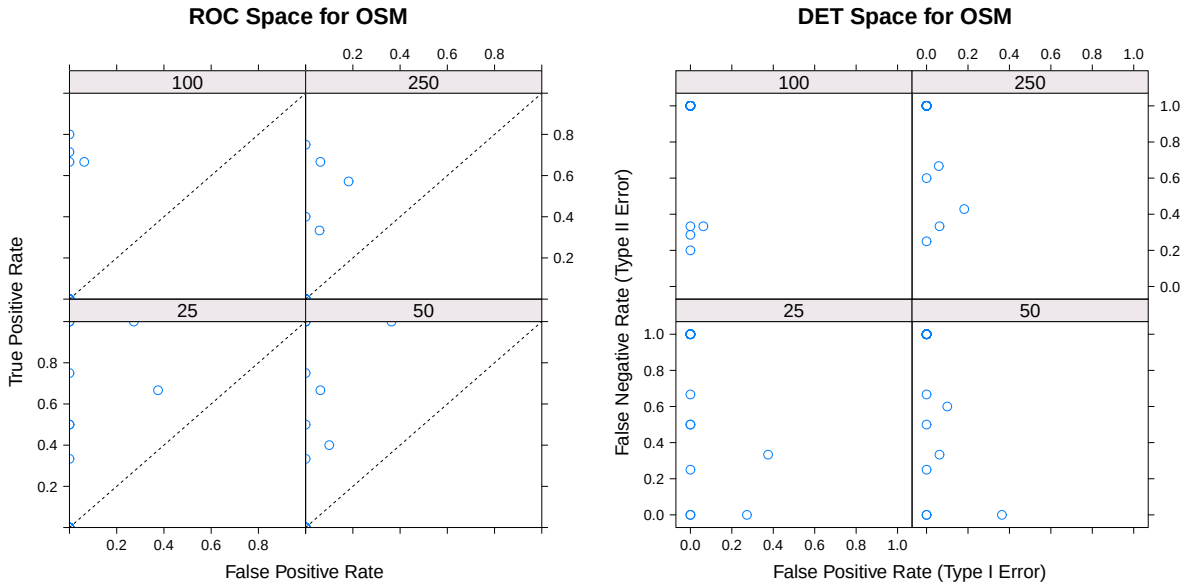
(a) Accuracy

(b) RMSE

(c) XV RMSE

(d) XV MSKV

Figure 7.17: Performance results for OSM data at different resampling lags using four metrics of interest.



(a) ROC (b) DET

Figure 7.18: ROC and DET curves for OSM measurements using “aggressive” resampling.

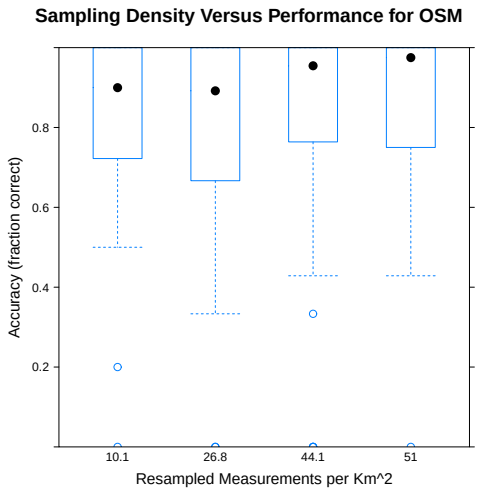


Figure 7.19: Hole prediction accuracy as a function of sampling density for OSM data.

state of the art *a priori* predictive methods or simple data fitting approaches. It also acheived nearly identical performance at predicting coverage holes to the sophisticated method of Robinson et al. [200], using the same number of measurements.

Next, crowd-sourced coverage mapping was investigated to understand the promise and practical limitations of a measurement methodology which requires no work from the network operators themselves. Although a simulation-based analysis of measurement coverage showed that only a small fraction of the population needs to participate to generate sufficient data, a case study using real measurements from the OSM project showed that in practice participation is too small to generate enough measurements for general purpose mapping. However, even with sparse measurements, the geostatistical mapping approach proposed in this thesis performs admirably. Although promising, a great deal more work is needed to understand the abilities and limitations of crowd-sourced coverage mapping in practice. Thankfully, from the perspective of further research, the data will just keep piling up on its own.

Chapter 8

Optimized Spatial Sampling

Previous chapters described a complete wireless coverage mapping based upon geostatistical methods. Chapters 5, 6, and 7 adapted standard geostatistical approaches to generate statistically robust coverage maps of wireless network coverage. Using experiences applying these methods to three different production networks using different technologies, best practices were derived for basic geostatistical methods when applied to this new domain. However, there is still room for improvement and this chapter attempts to take the next important step via optimized spatial sampling. By performing a secondary sampling phase aimed at fine tuning, spatial stationarity assumptions required by the methods described in the previous chapter can be relaxed, while also allowing for systematic tuning and regeneration of coverage maps over time.

The next sections will overview the state of the art with respect to optimized spatial sampling, discuss the approach taken here to adapt these methods to the domain of wireless networks, and provide results from case-studies using a production network.

8.1 Optimized Sampling

In addition to classic sampling systematic schemes, such as the equilateral triangular lattice used in the previous chapter, there has been some work on optimized spatial sampling in the geostatistics literature. Most works identify a conflict between the sampling needs of the two most important aspects of geostatistical modeling and estimation: variogram fitting requires samples at a variety of lags, and OK requires measurements to be largely uniformly distributed to minimize the distance between any two points. To resolve this conflict, most researchers select an initial sample similar to the one used in the previous

chapter: systematic grid-based sampling with some clustered measurements. Once sufficient information is gleaned about the underlying spatial process, an optimization strategy can choose subsequent measurement locations.

An early work in this area is [36], which describes the general problem of optimized sampling in two dimensions and shows that, unlike sampling in one dimension, there is no trivial solution. Instead, Bellhouse advocates a three-phase approach that involves (1) row-wise, (2) column-wise stratified sampling, and (3) independent (i.e., unaligned with the stratified ones). In [128], Lark investigates MLE-based sampling optimization via SSA. The authors advocate a pilot sample on a regular grid followed by a honing sample determined by their SSA algorithm. They also show that fields with small spatial correlation produce the most accurate models with clustered samples, while fields with large spatial correlation tend to prefer a regular grid-sampling. However, the results in that paper are based on simulated data and it is not clear how well the results are applicable to realistically noisy fields. In [139], Marchant and Lark expand on this work by proposing a metric to estimate model variance that is independent of the variogram estimation. Hence, they claim this metric (which is based on the Fisher information matrix) can be used as an optimization criteria for selecting samples both for Kriging and for variogram estimation. As with their prior work, all results are based on simulated values, and it is not clear how well these methods work for practical field estimation.

In [230], van Groenigan *et al.* propose a method for using SSA to select sample points by minimizing global Kriging variance. More advanced methods to optimize sampling account for nonstationarity, in time and/or space. Due to the clearly nonstationary nature of the wireless propagation environment, these methods define the direction taken in this chapter. In [93], Helle and Pebesma propose a similar line of research that also considers temporal variation (and placement of samples in time) in optimizing spatial sampling. In [40], Bueso provides a more complicated approach that is based on entropy maximization and uses it to study piezometric data from aquifers. In [63], Delmelle and Goovaerts present a complete two-phase approach that works by performing an initial sample and then choosing optimized second-phase samples. In that work, initial sampling is on a regular grid with augmented (clustered) random samples. See figure 8.1 for an example of 8x8 and 10x10 initial designs. After estimating the shape of the field using the initial

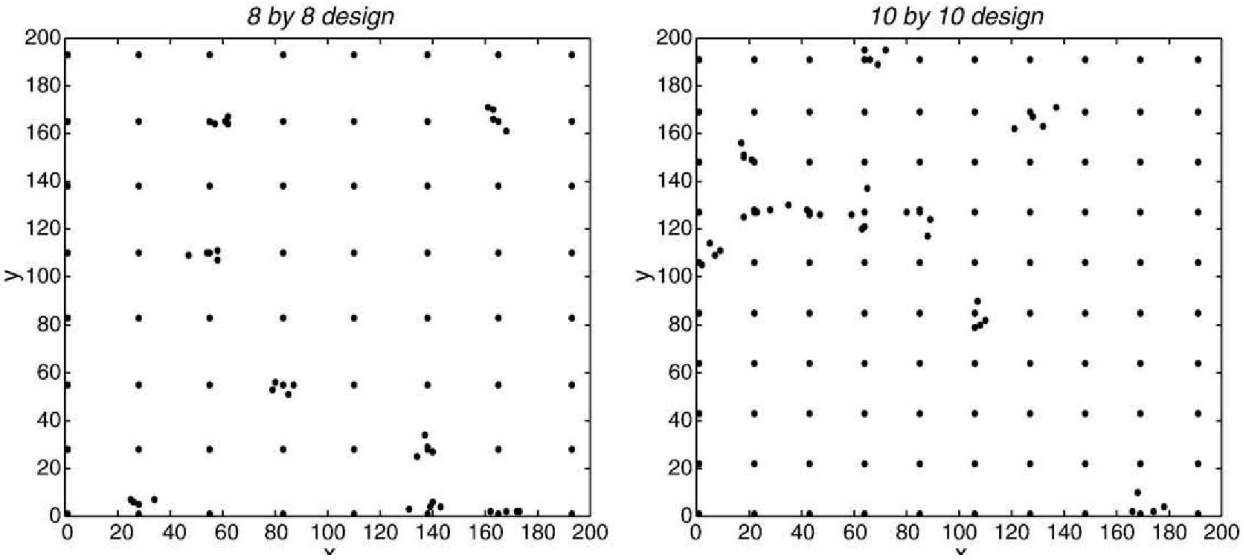


Figure 8.1: Systematic (grid) sampling augmented with nested ”random” samples. In the NxN case, points are placed on an equally spaced regular grid of NxN points and then N additional points are sampled in the immediate neighborhood of N grid points. This has the effect of creating an initial sample that both covers the region and has a range of lag distances between measurements for variogram estimation. Figure taken from [63].

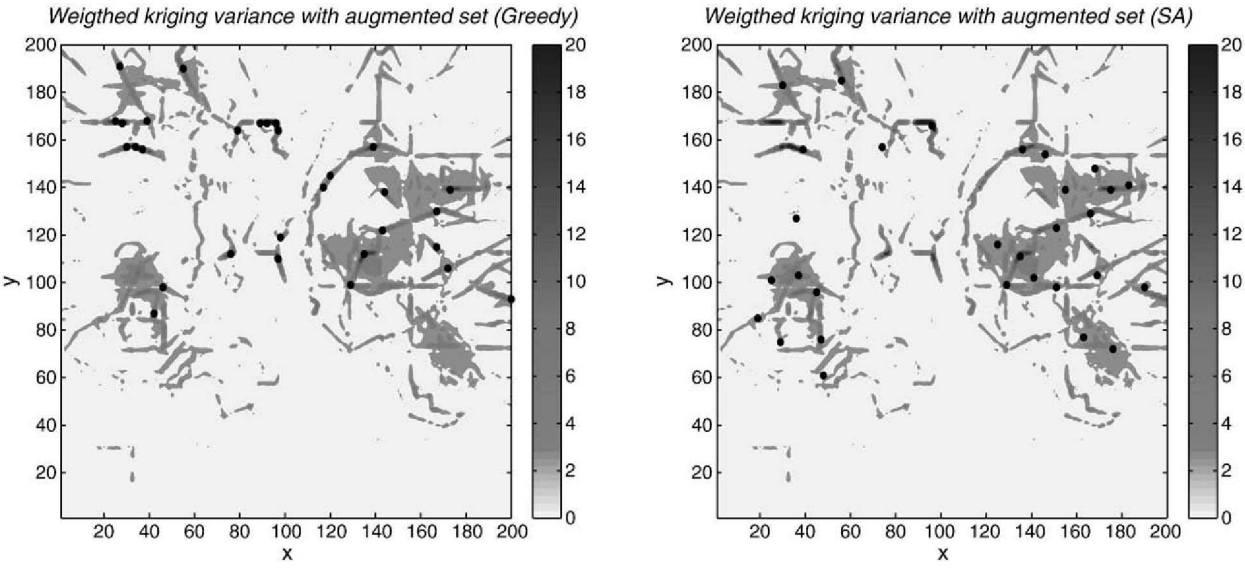


Figure 8.2: Optimized placement of second-phase samples according to the weighted Kriging variance proposed by Delmelle. Figure taken from [63].

sample, second-phase sampling sites are chosen by optimization. The authors investigate two optimization strategies: one uses a greedy approach, and the other is an SSA approach similar to that proposed by Lark *et al.* Delmelle also proposes a new optimization metric called "weighted Kriging variance" which accounts for abnormal variations (e.g., nonstationarity in space). Figure 8.2 shows where second-phase samples are placed based on optimizing this metric. They compare the various approaches using ground truth data collected from satellite imaging and show a clear relationship between the density of the initial sampling strategy and the gain from a corrective secondary strategy.

No matter the method, sample optimization is fundamentally combinatorial—each possible set of additional points must be analyzed and an optimal set found. Because there are infinitely-many possible additional points, an algorithm must choose to analyze a subset of candidates. Investigating only the best candidates first (pruning) is an active area of research. Although there has been substantial work on optimized sampling approaches in the geology literature, specifically in soil sampling, there has been no prior work applying these methods to radio propagation and wireless coverage. This chapter takes the multiphase nonstationary sampling approach suggested by [63] and attempts to refine the sampling and interpolation methods developed in previous chapters.

8.2 Method and Implementation

This section investigates methods for second-phase sample optimization. Initially, successively large simple random samples are generated to derive a lower-bound on gain from sampling as a function of number of samples. Using this lower bound, the relative gain from more complicated (optimizing) sampling methods can be evaluated. The methods described here are prototyped using a subset of the data collected in section 5. In particular, this chapter focuses on the problem of refining the CINR propagation map for the the GENI WiMax cuEN node (see table 6.1). The method developed assumes that some number of pilot samples have been taken, but does not make any assumptions about where they were made or of what metric. In the example of the cuEN node, the first-phase samples were taken on a 100m equilateral triangular grid over the CU campus.

8.2.1 Specification of Measurement Boundaries

In order to acheive a meaningful second-phase sample, the area of interest must be somewhat constrained. Doing so prevents the optimization algorithms from placing measurements where there is little available information (and hence, a large variance). Initially, experiments were attempted that constrained the second-phase sample to within the convex hull containing the pilot sample points (this hull is shown in figure 8.3). Despite having the benefit of being fully automatic, this method has some issues. Because the hull cannot closely fit the perimeter of the area of interest (the CU campus in the example), sample points will be preferred at edge locations where less information is available from the pilot sample. Although this is, in effect, the correct output from the optimization, it does not provide the most additional information in the areas within the hull, where the fidelity of the map is (presumably) most important.

As a more laborious but effective alternative, a method involving manual specification of boundaries using Geographic Information System (GIS) software was chosen. Figure 8.4 shows two sets of geospatial polygons defined using the Quantum GIS open-source software [177]. The first set defines the perimeter of the area of interest, and second-phase sample points are allowed only within this polygon. The second set of polygons define "unmeasureable" areas, which are mostly buildings, bodies of water, and construction zones (although it may be possible to make measurements in these locations, mapping *accessible* outdoor coverage is the aim of this thesis). Defining the polygons manually using the GIS software is a straightforward task, which involves placing boundaries on top of United States Geological Survey (USGS) orthoimagery. Although this is a manual process, it can be done quickly. For instance, the polygon sets used in this analysis were defined in less than an hour of work. These polygon sets are exported from the GIS software as a pair of Environmental Systems Research Institute (ERSI) shapefiles [2]. Testing a point for inclusion in a polygon in either of these shapefiles is accomplished quickly using a tool developed using the shapelib C library, and shapely Python library [138, 79].

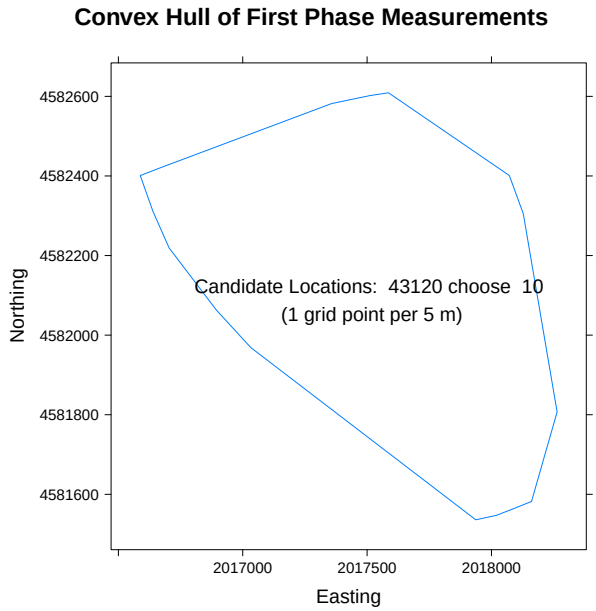
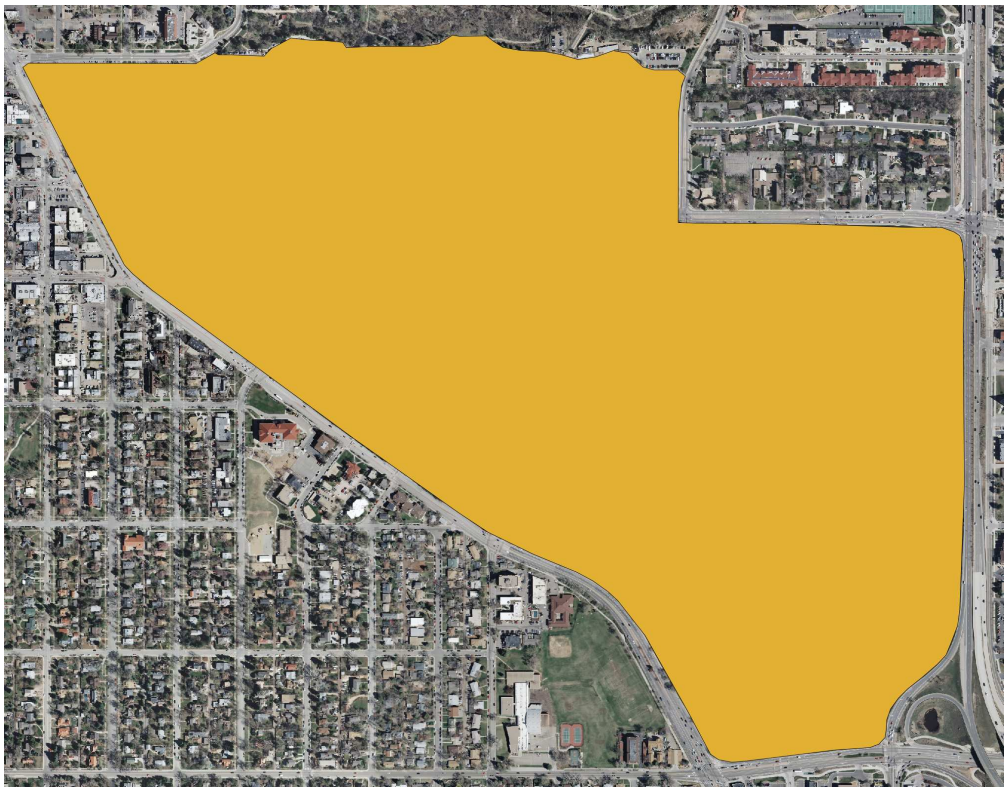
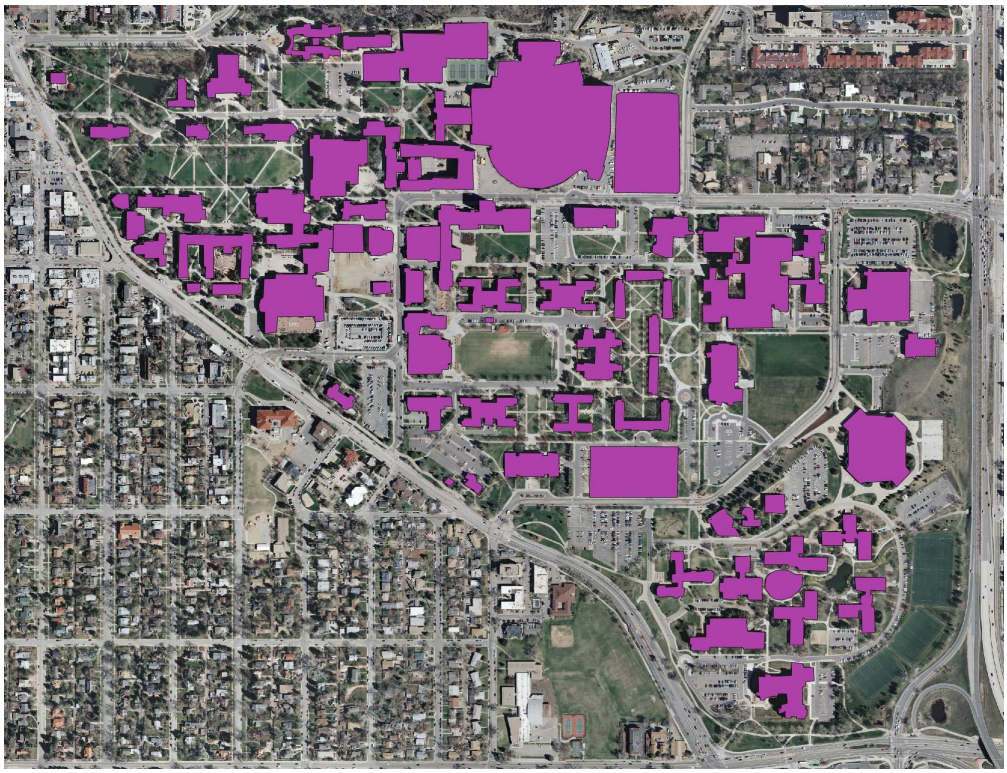


Figure 8.3: Convex hull around first-phase sample points for cuEN node.



(a) Measurement Boundary



(b) Unmeasureable Locations

Figure 8.4: Manual specification of boundary and unmeasureable polygons using the Quantum GIS software.

8.2.2 Metrics of Sampling Gain

Following the method of [63], two metrics are used to analyze sampling gain: Average Kriging Variance (AKV) and Weighted Kriging Variance (WPE). AKV is the average Kriging variance of a given sampling design. It can be computed at a given point ($\mathbf{x}$) using the (fitted) variogram and the inverse of the variance-covariance matrix (K^{-1}) [197]:

$$\sigma_E(\mathbf{x})^2 = \sigma^2 - k(\mathbf{x})^T K^{-1} k(\mathbf{x})) \tag{8.1}$$

where σ^2 is the fitted variogram sill parameter, $k(\mathbf{x})$ is the value of the covariance function at point $\mathbf{x}$ (a vector with one value for each other point), and $k(\mathbf{x})^T$ denotes the transpose of $k(\mathbf{x})^T$ [1]. Computing this value is nontrivial. The implementation developed here is given in appendix C, section C.1.

Taking the arithmetic average for all $\mathbf{x}$ produces the AKV metric:

$$AKV = \frac{1}{N} \sum_{i}^{N} \sigma_E(\mathbf{x}_i) \tag{8.2}$$

where N is the number of candidate locations (grid/pixels) in the map. N varies depending on the resolution desired for the Kriged map, which determines the resolution available to the sampling algorithm. Unless stated otherwise, this chapter will use a resolution of 0.2 pixels per meter (or one pixel every $5m^2$), as was done in prior chapters.

The second metric, WPE, is a roughness-weighted Kriging variance. In [63], Delmelle and Goovaerts describe a method to compute a spatial roughness matrix on the Kriged map that is used to scale the Kriging variance matrix. Because Kriging variance is a function of the fitted variogram and the positions of locations, and not the actual sampled values, Delmelle and Goovaerts argue that the AKV metric alone misses important sampling regions in nonstationary processes where there is substantial change in interpolated value ("roughness") over short distances. The roughness at a given point $\mathbf{x}$ is defined as:

[1] [233, pp.12-19] provides an excellent discussion of how the variance-covariance matrix is derived and the meaning of the spatial covariance function.

$$r(\mathbf{x}) = \sum_{j \in J} \frac{d(\mathbf{x_j}, \mathbf{x})^{-\beta} (Z(\mathbf{x_j}) - Z(\mathbf{x_j})^2}{\sum_{k \in J} d(\mathbf{x}, \mathbf{x_k})} \tag{8.3}$$

where J is the set indexes of points in the neighborhood of $\mathbf{x}$ (but excluding $\mathbf{x}$ itself), $Z(\mathbf{x})$ is the Kriged (interpolated) value at $\mathbf{x}$, $d(\mathbf{x}, \mathbf{y})$ is the euclidean distance (in meters) between the points $\mathbf{x}$ and $\mathbf{y}$, and β is a constant parameter chosen to weight the distance-scaling. The neighborhood size J and parameter β are chosen by the experimenter. In this work the precedents of Delmelle *et al.* are followed, with $\beta = 1.5$ and a J that includes the eight grid points that immediately surround the point $\mathbf{x}$. The $r(\mathbf{x})$ function is used to calculate WPE as follows:

$$WPE = \frac{1}{N} \sum_{i}^{N} \left(\frac{r(\mathbf{x_i})}{R} \right)^{\alpha} \sigma_E(\mathbf{x})^2 \tag{8.4}$$

where R is the maximum $r(\mathbf{x})$ value for all $\mathbf{x}$, which is used to normalize the $r(\mathbf{x})$ values. α is a paramter that controls the importance of the roughness weights. In this chapter, $\alpha = 1$ is used, following the method of Delmelle et al.

8.2.3 Simple Random Sampling

The most straightforward approach (in terms of analysis and design) to spatial sampling is simple random sampling. In order to determine a lower-bound for gain associated with second-phase samples, a simple random sample is taken within the convex hull of first-phase measurement points. Figure 8.5 shows the reduction in WPE and AKV for increasingly large random samples. To generate this figure, increasingly large random samples in increments of 10 up to 1000 are selected at new sample locations: $n = 10, 20, ..., 1000$. At each value of n, ten unique random samples were generated to get an estimate of central tendency and spread. This results in 1000 unique uniform simple random samples, for which the two metrics, AKV and WPE, are computed. The reduction in these metrics has a clear inverse logarithmic shape with increasing random samples, and a horizontal asymptote around 1.5 for both AKV and WPE. A reduction of ≈ 0.25 is acheived after 30 samples, ≈ 0.5 after 100 random samples, ≈ 1.0 after 400, and ≈ 1.5 after nearly 1000. More sophisticated (optimized) sampling strategies are investigated in the next

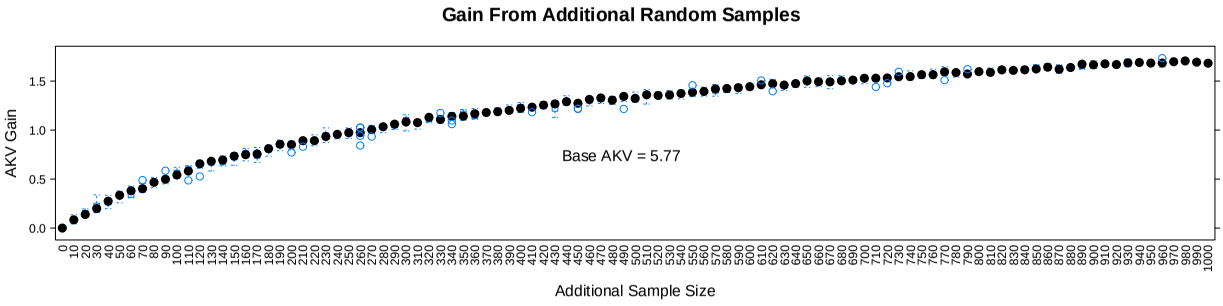
subsections, and their efficacy will be described relative to these curves—a reduction in gain that is better than random at a given sample size is a success. Determining how great of a reduction is possible in practice is the question the remainder of this chapter will seek to address.

8.2.4 Second Phase Greedy Optimized Sampling

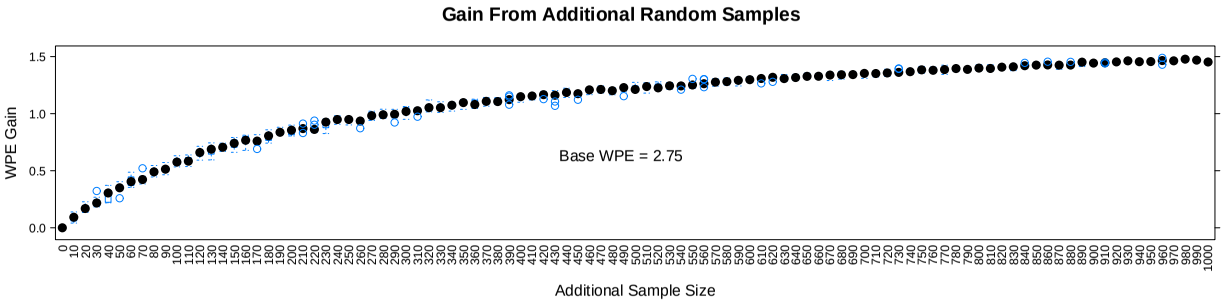
In general, the goal of the sampling optimization process is to select a random sample of size M that most reduces the metric of error, AKV or WPE. With N candidate locations, the complexity of this problem is combinatorial: $\binom{N}{M}$. Solving this problem exactly is intractable for any reasonably sized N and M. For instance, with $N = 31,056$, which is the number of cells on a five meter grid within the bounding polygon and excluding the "unmeasureable" polygons, selecting a new sample of M points may require inspecting as many as $2.65185204 \times 10^{38}$ options. To resolve the fundamental intractability of this problem, greedy and stochastic methods are the approaches investigated here. In this section, a greedy (myopic) method that iteratively chooses the point with the highest WPE value on the map is used. Although this seems like a reasonable approach at first glance, it generates samples that are heavily skewed towards local maxima.

Figure 8.6 shows the maps for the cuEN node, including the original Kriged detrended map (as computed using the methods described in the chapter 5), the residual standard error (square root of Kriging variance), roughness as per equation 8.3, and WPE (roughness scaled error/variance). To determine the efficacy of the greedy method, 100 greedy samples were computed by choosing the point (pixel) with the highest WPE value, then recalculating the WPE map and choosing the highest WPE value again. Figure 8.7 shows the improvement in AKV and average WPE as a function of increasing samples. There is a total reduction of approximately 0.11 dB in AKV and 0.04 WPE after 100 samples. The curve appears to flatten substantially after 20 samples, when approximately 90% of the total improvement has been gained. This result compares unfavorably to the random approach described in the previous section which produces greater improvement, presumably because the random samples tend to spread points over the entire region of study.

Despite succeeding in providing a modest reduction in the overall error (and roughness), this greedy approach produces a myopic sample that is unlikely to provide much useful additional information to the

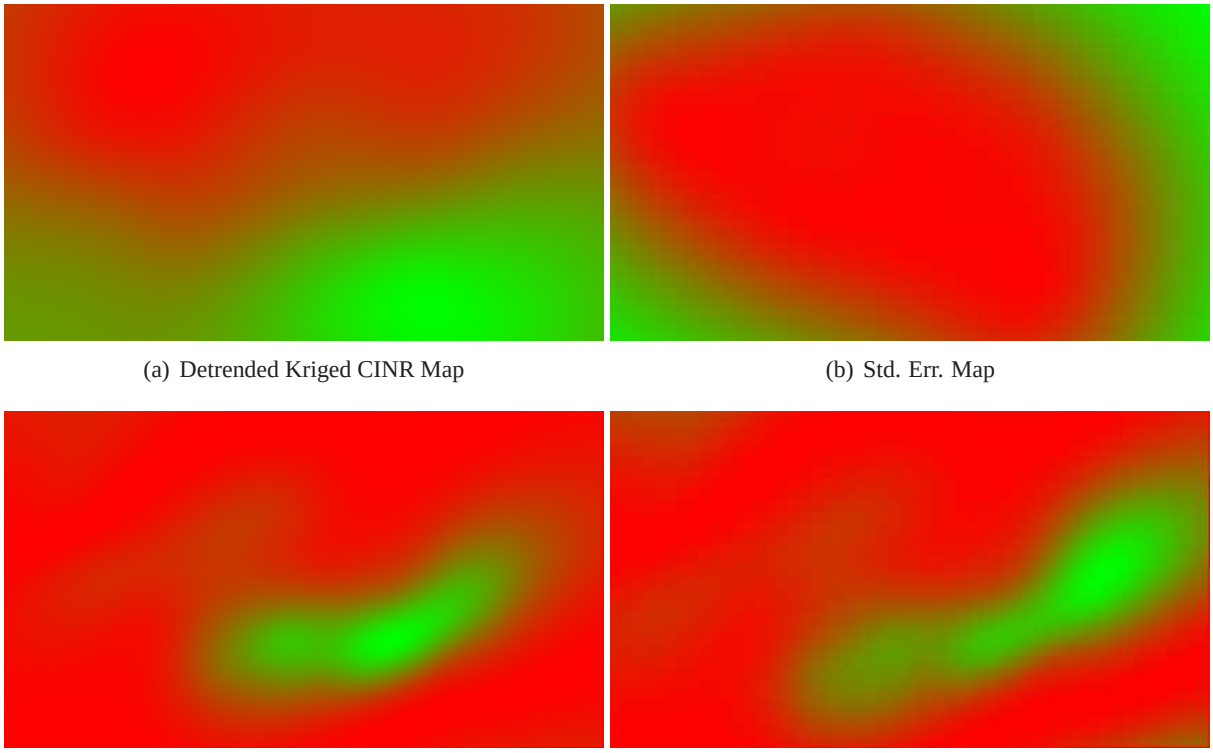


(a) AKV Gain



(b) WPE Gain

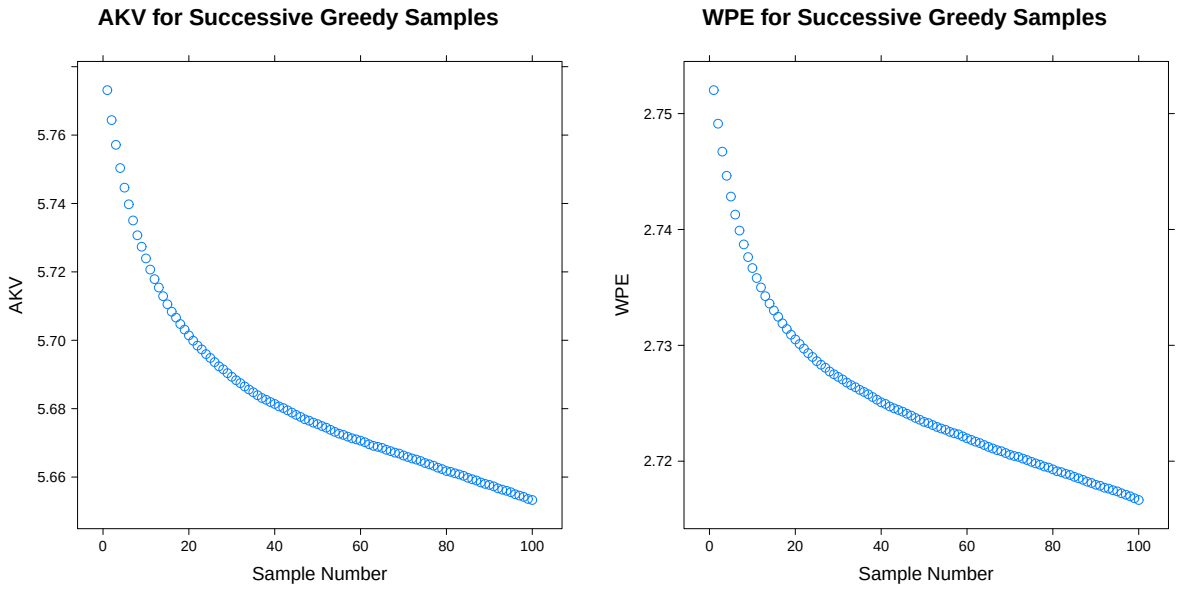
Figure 8.5: Performance gain from increasingly large simple random samples.



(a) Detrended Kriged CINR Map (b) Std. Err. Map

(c) Roughness (d) WPE Map

Figure 8.6: Maps used for sample optimization for cuEN node.



(a) AKV (b) WPE

Figure 8.7: Improvement (gain) from iterative greedy sampling using the WPE map.

mapping process. Figure 8.8 shows the resulting sample. The greedy approach simply places all points as near as possible to the the highest value point on the WPE map. Although this will certainly reduce the error in that region, it does not provide useful information about any other locations (that may contribute to the overall error) on the map. The next section will attempt to resolve this problem using a stochastic optimization algorithm.

8.2.5 Second Phase Spatial Simulated Annealing Optimized Sampling

Basic Simulated Annealing (SA) is a classic stochastic "metaheuristic" search algorithm originally proposed by Metropolis *et al.* to mimic the annealing of metal [144]. SA has been used effectively in a great number of problems, the closest of which to sample optimization is probably the problem of location planning, where a fixed number of "resources" are placed geospatially to satisfy demand and the constriants of the system [147]. In SA, random solutions are generated and tested. If a random solution is the best seen, it is kept. If, however, the new random solution reduces the overall fitness, then it is kept with a probability determined by the Metropolis equation:

$$Pr[accept] = e^{\frac{\Delta f}{t}} \tag{8.5}$$

where Δf is the change in fitness (which is necessarily positive) and t is the current "temperature" of the system. The temperature is reduced each iteration. In the implementation used here, the temperature begins at 1.0 and is reduced linearly thereafter.

This algorithm is meant to accept bad (fitness reducing) moves with a greater probability at first, and then less frequently after many iterations (as the temperature cools). The range of the values of Δf and t can drastically effect the way this plays out in practice. In the implementation used here, it is assumed that t ranges from 1 to 0 and that Δf from 0 to 1 as well. Figure 8.9 gives a heatmap of the probability of accepting a bad move as a function of the size of Δf and t. In practice, fitness values are actually much too small for this to work, and are generally in the neighborhood of 0.02 and sometimes much smaller. To get the desired effect, Δf is scaled up by a factor of 100. In general, the appropriate scaling factor could

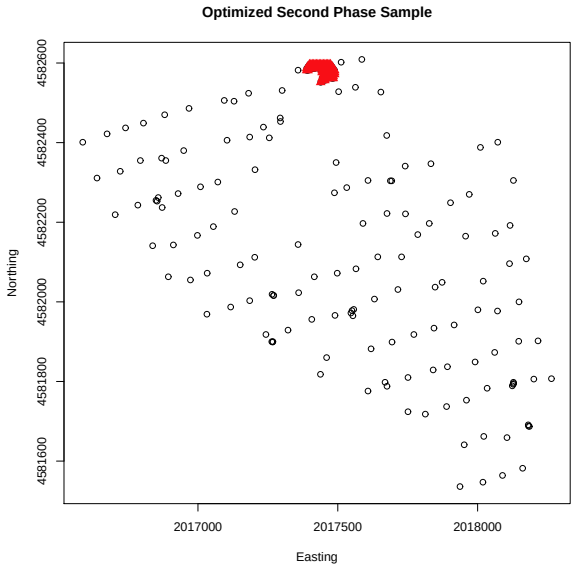


Figure 8.8: Greedy sample of 100 points. Preexisting sample locations are denoted with circles and second-phase (greedy) samples are filled (red) triangles.

be determined automatically by generating a population of random solutions, determining the distribution of Δf values between them, and then scaling the range to fit the expectation of $[0, 1]$.

The SSA extension to basic SA is straightforward: a list of candidate sample locations is generated on a uniform grid within the spatial constraints defined by the boundary polygons. The density of this grid effects the granularity of the search space, and hence the solutions produced by this method. For the sake of evaluation, 0.2 pixels (grid points) per meter is used, which is also consistent with previous experiments in chapters 5, 6, and 7. Next, the sample size (N) and number of iterations to use (M) are chosen. For evaluation purposes, N of 10, 25, and 50 point samples are used with $M = 1000$ iterations. This tends to agree with the soil sampling work in [63], where the authors generate samples of size 30 and allow the SSA algorithm to run for 850 iterations. At the first iteration, a sample of N points are randomly selected from the list of candidates. Then, with each successive iteration, a new sample is generated by replacing one point in the current sample with a random (unused) candidate. This new, altered sample is kept if it is an improvement; if it is not, whether it is kept is determined by equation 8.5.

Figure 8.10 shows the value of AKV and WPE for successive iterations. It should be noted that these plots show the *improvement over a random sample*, since the starting state is a random sample which is iteratively optimized. For the case of $N = 10$ and $N = 25$ an improvement in WPE of 0.3 and approximately 0.25 is obtained, respectively. Looking at the placement of the resulting samples in figure 8.11, it is clear to see that this approach generates a more reasonable sample than the greedy sample. In both cases, the new points have been located at areas of interest, particularly around edges of the coverage region, and in the back-lobe of the antenna propagation, where the signal observed was weak during initial measurement. The SSA algorithm manages to select these sample locations simply by trying to minimize the residual error and roughness of the geostatistical model without any information about the antenna geometry or environment.

Another example with $N = 50$ is given in figures 8.12 and 8.13. In this experiment, the placement of 50 additional second-phase samples is obtained, with increasing numbers of iterations. With a starting temperature of 1000 (i.e., 1000 iterations), there is an improvement in the WPE metric of 0.25 over random. With 2000 iterations, a gain of approximately 0.35 is acheived. After 5000 iterations, the gain is nearly 0.5, and after 10,000 iterations (which takes many days to compute on a single computer), the gain is still

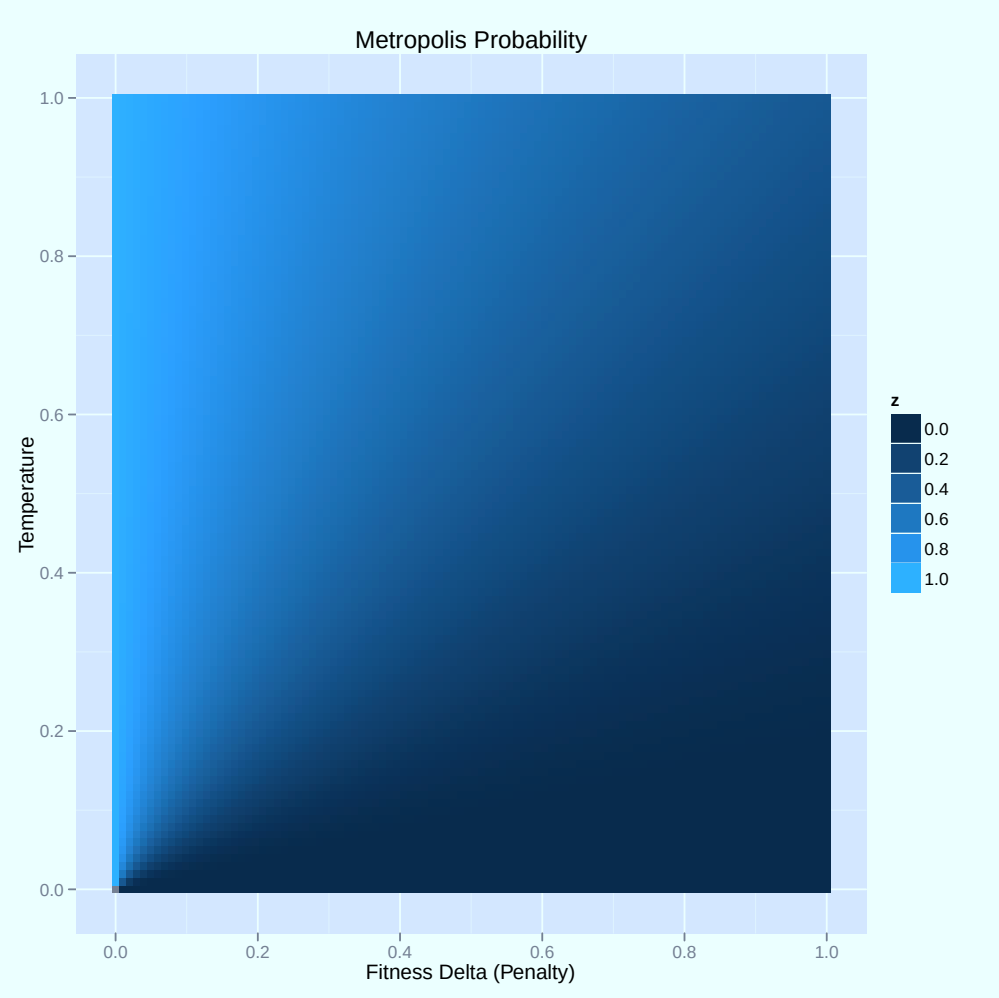
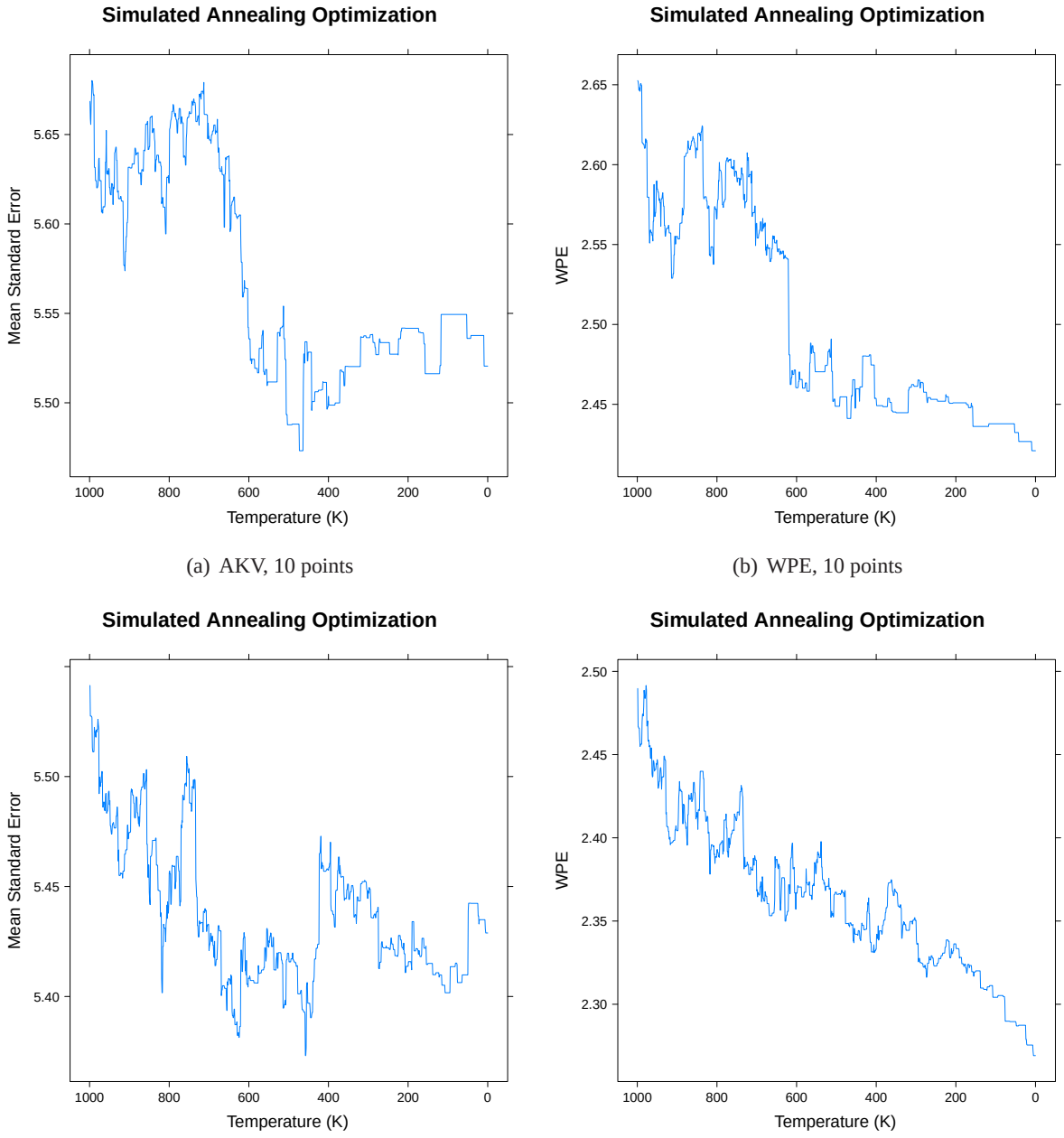


Figure 8.9: Metropolis probability plot for spatial simulated annealing.

0.5. This result implies that there is an asymptote around 2.0 WPE for this particular scenario and method and that additional iterations are not worthwhile. The samples chosen by this method are shown in figure 8.13. As with the results from the experiment with $N = 25$ discussed above, the optimized samples are positioned to focus additional measurements in the mainlobe and backlobe of the transmitter, around its base, and amongst gaps in the spatial sample. There is a meaningful difference between the 1000 iteration scenario and the 2000 iteration scenario, but little difference with increased iterations. There appears to be a maximum obtainable gain for a given scenario that can be obtained with sample optimization.

A final example is given in figure 8.14, which shows equivalent plots to the plots described above, but utilizing the ESNR metric instead of the CINR metric. The ESNR metrics choose to place the points in similar locations to those optimizing on CINR, prioritizing measurements around near the edges with a few centered at coverage boundaries. Understanding whether there is a qualitative difference between which metric is optimized is a topic for future work.



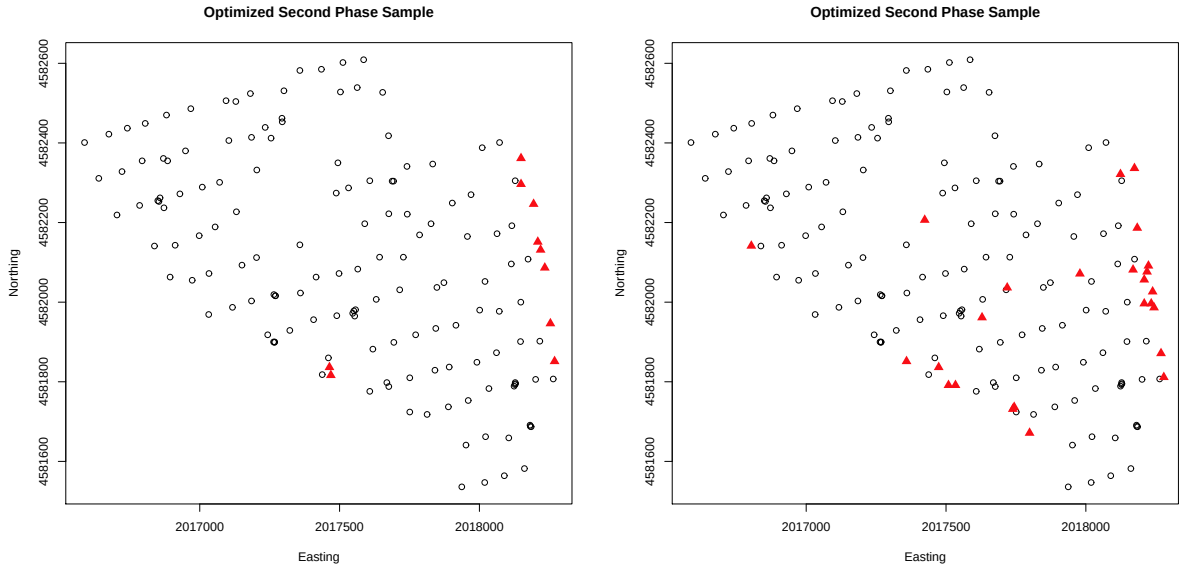
(a) AKV, 10 points

(b) WPE, 10 points

(c) AKV, 25 points

(d) WPE, 25 points

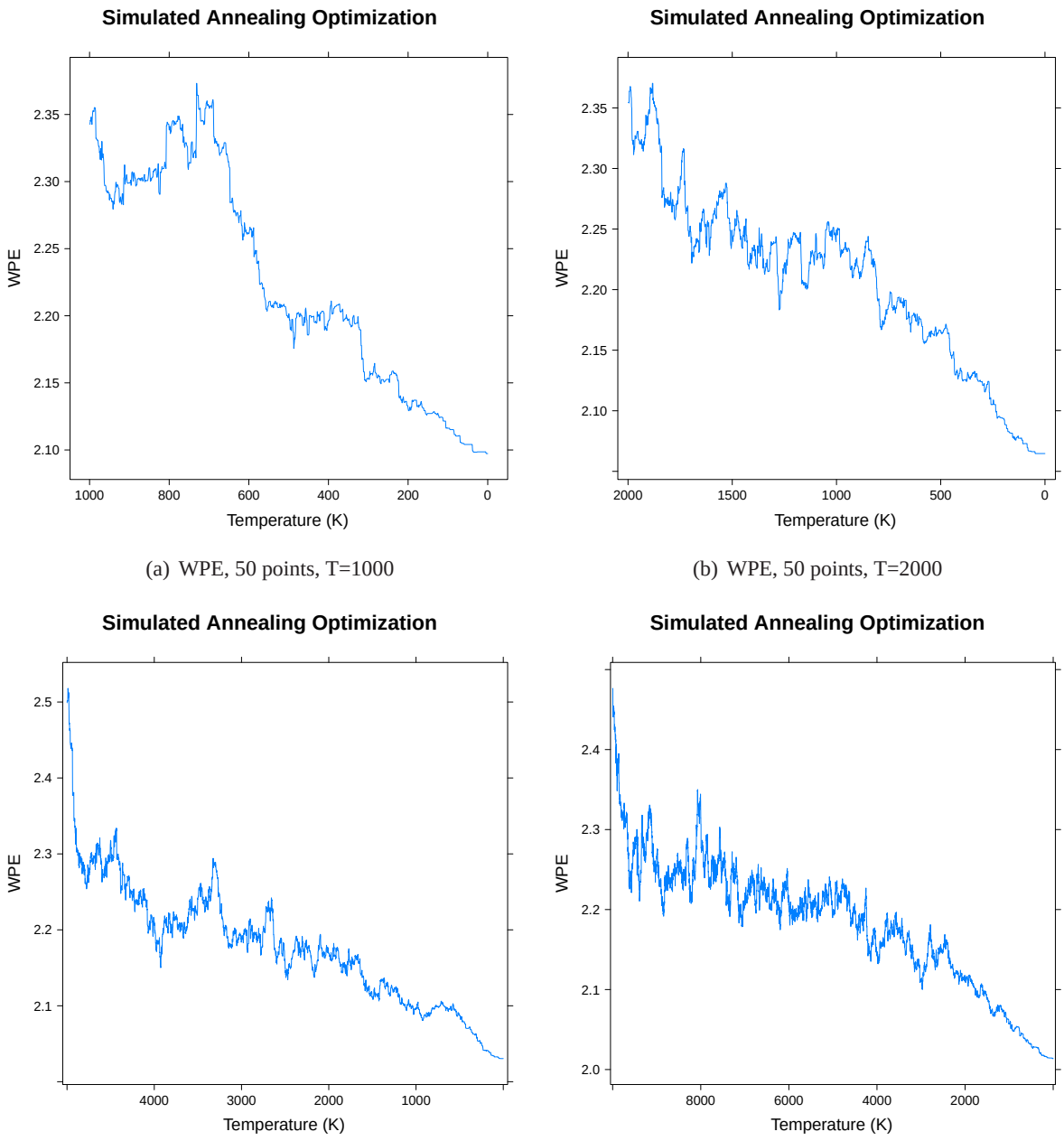
Figure 8.10: Improvement (gain) from spatial simulated annealing using the WPE map for sets of 10 or 25 additional points.



(a) Sample, 10 points

(b) Sample, 25 points

Figure 8.11: Optimized sample for 10 and 25 points after 1000 iterations.



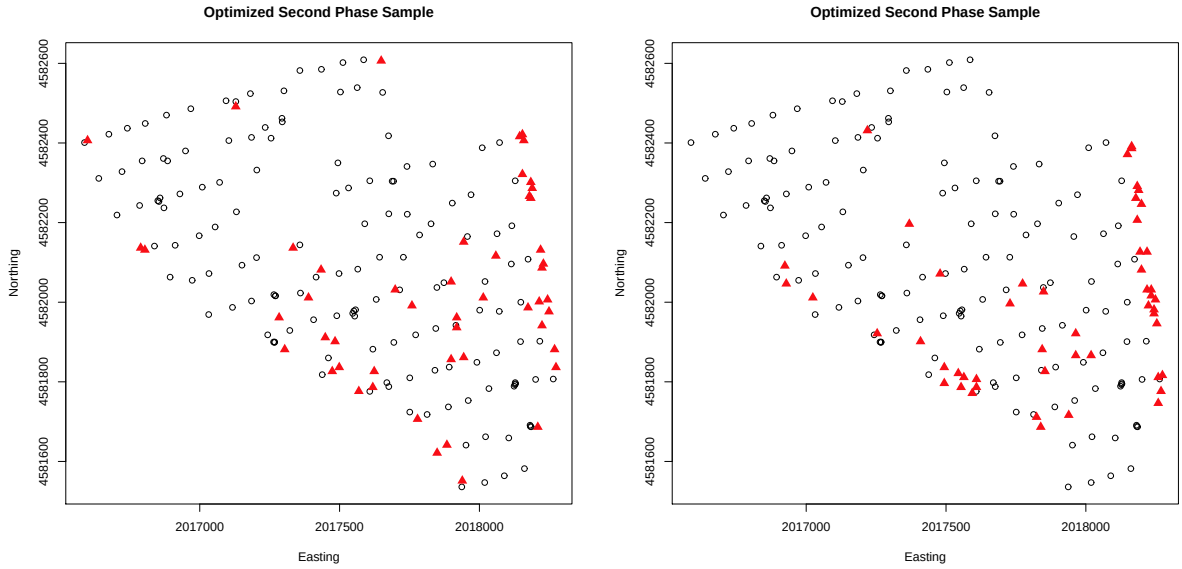
(a) WPE, 50 points, T=1000

(b) WPE, 50 points, T=2000

(c) WPE, 50 points, T=5000

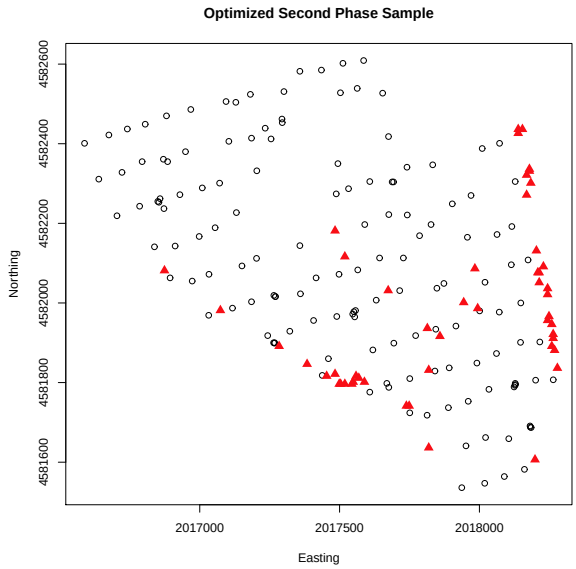
(d) WPE, 50 points, T=10000

Figure 8.12: Improvement (gain) from spatial simulated annealing using the WPE map for 50 additional points with increasing numbers of iterations

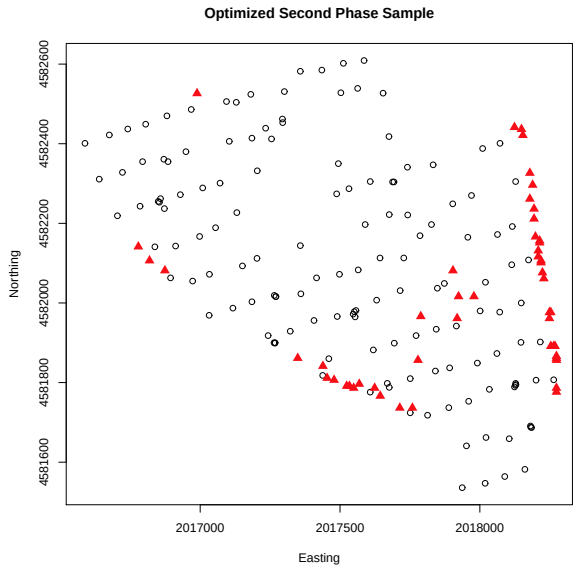


(a) WPE, 50 points, T=1000

(b) WPE, 50 points, T=2000

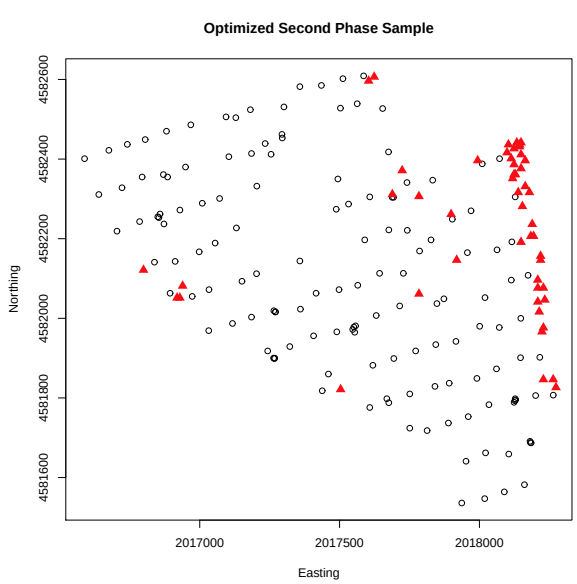


(c) WPE, 50 points, T=5000

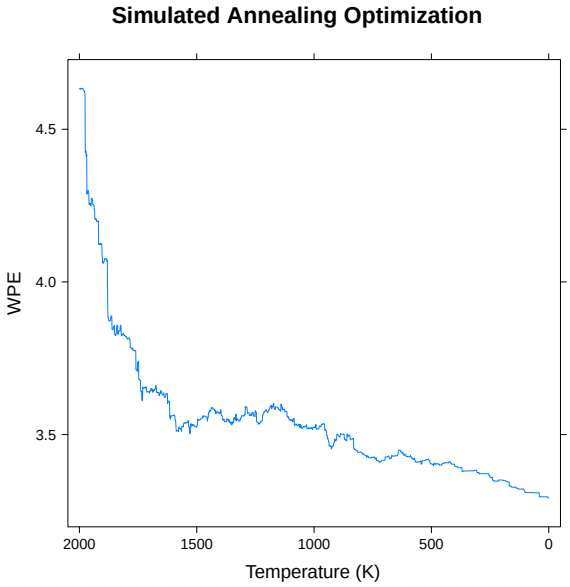


(d) WPE, 50 points, T=10000

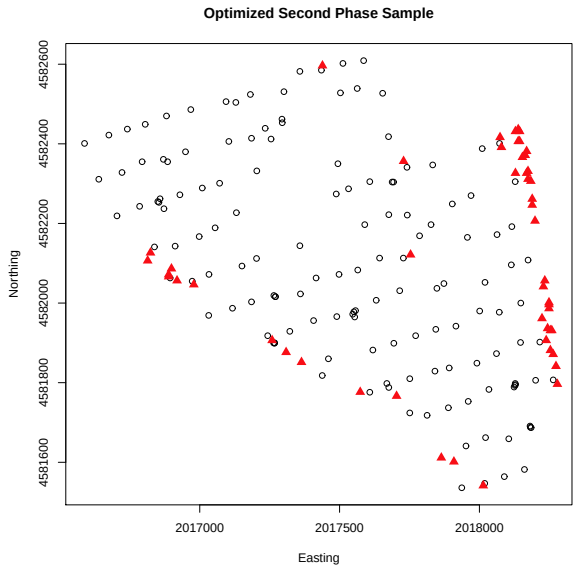
Figure 8.13: Optimized sample using the WPE map for 50 additional points with increasing numbers of iterations



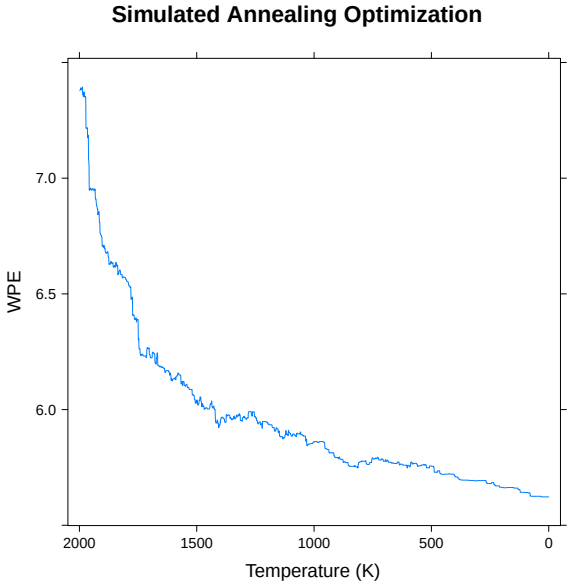
(a) ESNR6, 50 points, T=2000



(b) ESNR6, 50 points, T=2000



(c) ESNR54, 50 points, T=2000



(d) ESNR54, 50 points, T=2000

Figure 8.14: Optimized sample using the WPE map for 50 additional points with increasing numbers of iterations, but with the ESNR metrics.

8.3 Parallel Spatial Simulated Annealing

Spatial simulated annealing is a stochastic "hill-climbing"-style search strategy. This section makes the observation that this process can be parallelized such that a number of threads can perform a search simultaneously, which linearly increases the amount of the search space that can be explored in a given amount of time. Because stochastic search methods rely on "stumbling" into a good solution, this has the effect of multiplicably increasing the likelihood of a good move. To accomplish this, we follow the parallel metaheuristic approach described in [32], termed Multiple Independent Runs (MIR)-style SA. In this approach, a pool of candidate solutions is maintained from which a number of worker threads pick one, perform a sequential simulated annealing search, and then place the optimized solution back in the pool. After some number of runs, the best solution in the pool is accepted.

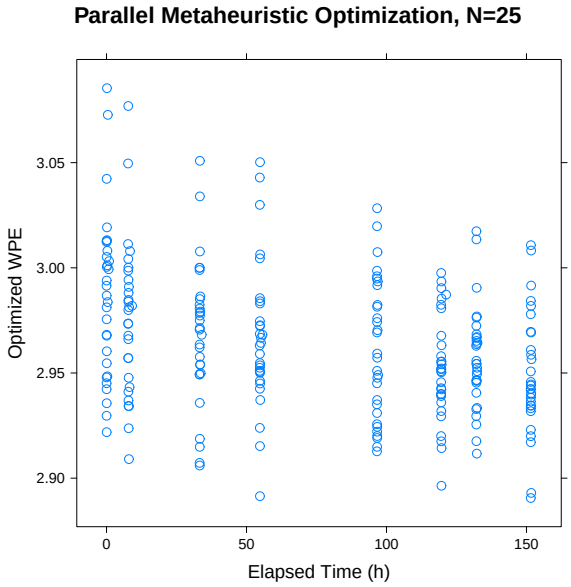
The first implementation tried makes use of 30 workers and a pool of 50 candidates. At least 200 runs are allowed to complete before choosing the winner, which results in approximately 6 runs per worker and 4 runs per candidate. As before, the initial temperature of the SA algorithm is set to 1000, which results in as many iterations per run. Despite being the standard approach to parallel metaheuristic optimization, this construction proved to be inaffective here, primarily because there is little gain from re-optimizing a given sample. For instance, figure 8.15 shows the fitness of optimized solutions in the pool over the course of the experiment with $N = 25$ sample points. In order to obtain higher throughput and fault-tolerance on the Janus supercomputer, a checkpointing mechanism was developed. This way, the compute time per run is reduced to approximately 5-10 hours, progress is saved, and then a second run is queued. In the time-series, the gaps are the periods of time where the job was queued, awaiting allocation on the cluster. In this time-series, a clear downward trend is present in the center optimized value. However, the minimum value appears to be fairly constant after approximately the fourth queue slot. This observation is clarified in the second subfigure, which shows the WPE gain over time. The distribution of gain appears to be roughly bi-modal. Over the first several runs, all of the random unoptimized candidates are tried and optimized, which produces a gain in the neighborhood of 0.7. Then, those optimized solutions are successively re-optimized as they are selected again and again from the pool. The re-optimizations produce much more modest gains,

which can be seen centered around zero. After about 100 hours, or 150 slave runs (30 per run, 5 queue runs) all of the low-hanging fruit have been picked from the pool, and the remaining gain to be extracted is marginal.

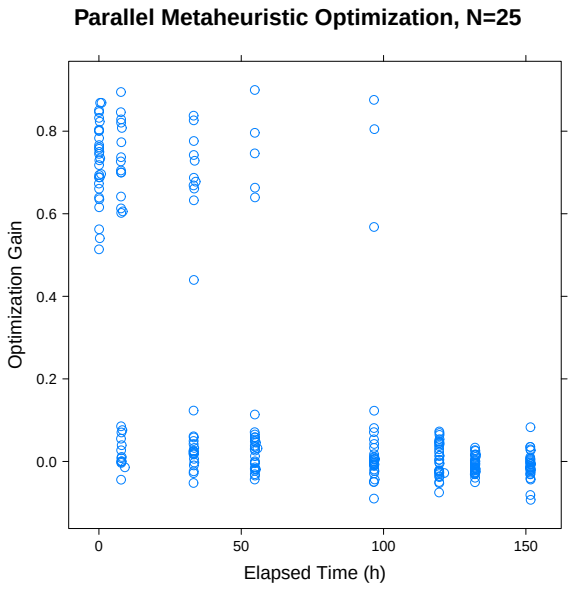
Figure 8.16 draws out this observation by showing the distribution of gains is clearly bimodal. In addition to the top figure, which shows the overall distribution, the bottom figures show the distribution of the biggest gains, which have a mean of 0.7, and the small gains, which have a mean of 0.01 (and sometimes are actually negative). From this experiment, it is clear that re-optimization of samples is not terribly fruitful. Instead, the greatest benefit of parallelization appears to be the ability to try a large number of initial random candidates. From figure 8.16(b), a fairly large spread of gain from the initial optimization of candidates is visible, running from 0.5 to 0.9. Based on this experiment, the decision was made to increase the size of the pool and allow only one run per candidate, which results in a large number of identical optimizations from random starting points.

To achieve greater parallelism on each compute node, calculation of the Kriging variance matrix can also be parallelized. The parallel function implementation in R is provided in appendix C, section C.1. For matrices of the size required here, approximately eight simultaneous threads are able to reduce the computation of this matrix by about three times. A modest increase in the number of threads performing this task will decrease the computation time, but above and beyond that, the task becomes communication-bound, and the performance gain is lost. Hence, *at least* 30 simultaneous workers are run, each of which itself uses between 8 and 12 simultaneous threads.

For these experiments, the Janus supercomputer is used, which has 1368 compute nodes total, each of which has 12 Central Processing Unit (CPU) cores and 2 GB of Random Access Memory (RAM) per core [156]. The OpenMPI message passing library and a master/slave architecture are used [220]. The master is responsible for maintaining a pool of candidate solutions, providing them to worker threads, and receiving and categorizing the optimized solutions when they are available. Because the Janus system prioritizes wide jobs, in most cases it is more efficient to avoid checkpointing and instead start 201 workers (200 slaves plus the master), each using 12 cores to parallelize the Kriging variance computation, which amounts to a staggering 2412 cores and 4.71 TB of RAM allocated to the job.

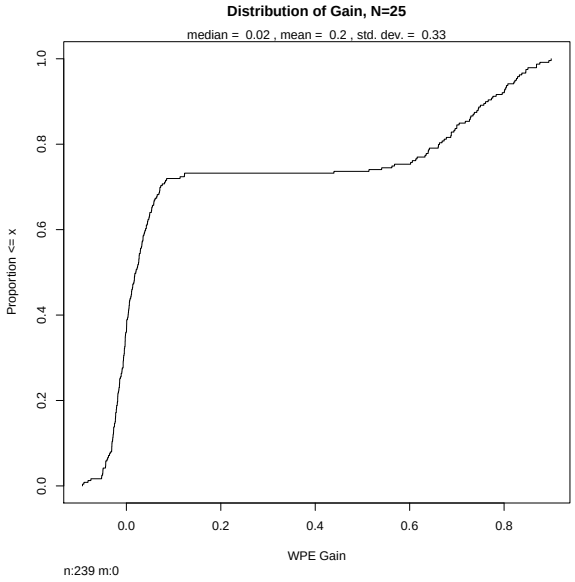


(a) WPE



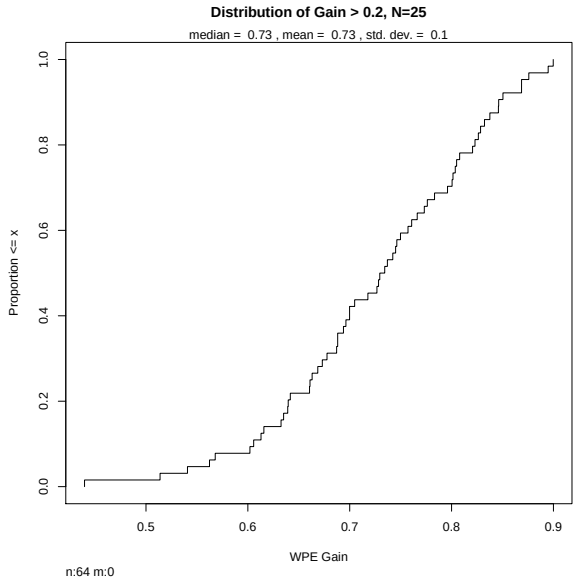
(b) WPE Gain

Figure 8.15: Optimization timeseries for $N = 25$.

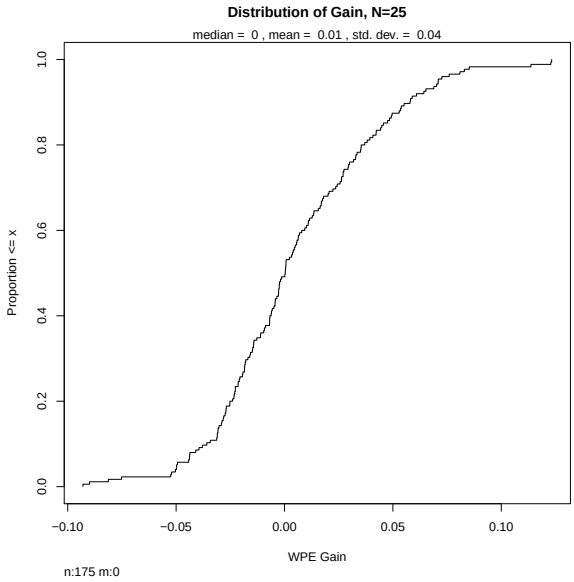


(a) Complete Distribution

(b) Thesis Waldo



(c) Upper End



(d) Lower End

Figure 8.16: Distribution of gain for $N = 25$. Register your sighting of Thesis Waldo at thesiswaldo.org. Waldo image by Martin Handford, distributed by Classic Media Distribution Ltd..

8.3.1 Experiment 1: N = 10

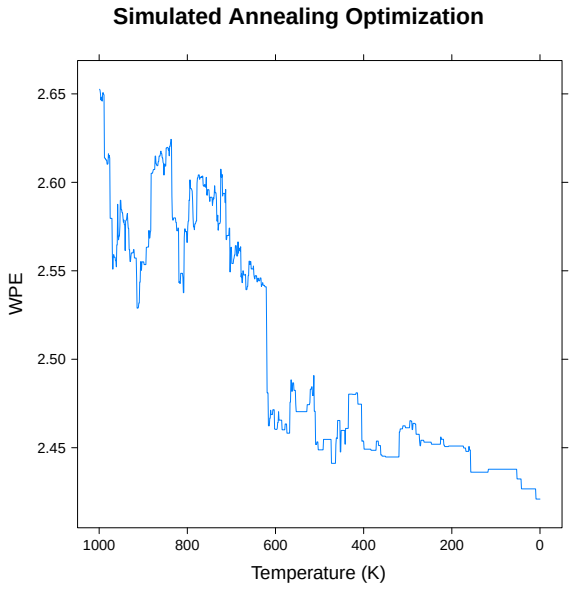
Figure 8.17 provides a comparison of the optimized second-phase sample for $N = 10$ using sequential SA and parallel SA. For the parallel version, the result from the best optimizing run is given. The parallel version acheives a small decrease the WPE metric of approximately 0.02 over the sequential version, which obtains a value of approximately 2.4. Both solutions choose to place all of the points near boundaries of the campus where the largest gaps in the grid-based measurements exist. There appears to be little qualitative distinction between the two solutions and the conclusion from this experiment appears to be that the parallel approach to optimization provides no meaningful gain over the serial approach.

8.3.2 Experiment 2: N = 25

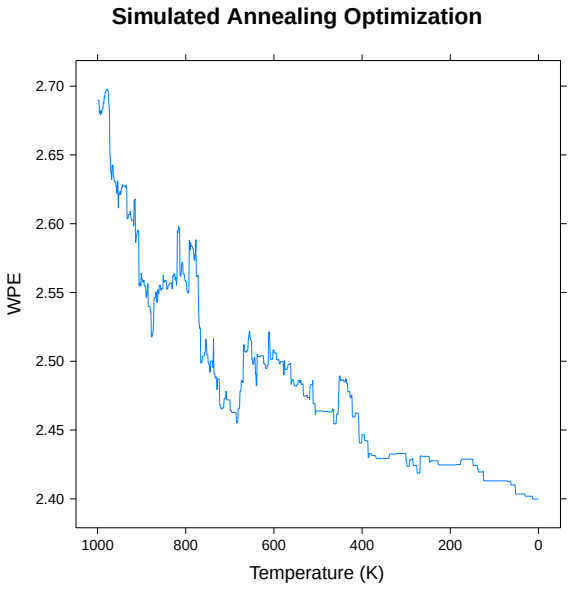
In the next experiment, $N = 25$ to optimize 25 sample points. Figure 8.18 shows the performance and final sample for the best and worst optimized solution. The worst optimized solution acheives a WPE of 2.30 and the best, 2.23 (a gain of approximately 0.25 and 0.32 over a random sample, respectively). Compared to the sequential solution, which produced an gain of approximately 0.25 (average among the pool), the benefit of parallelization appears to be an additional 0.07 gain in the best case. However, despite this quantitative affirmation for the parallel process, the produced sample, given as subfigures c and d in figure 8.18, tells a different story. These samples are not largely different qualitatively—they both place additional sample points in the same three regions as does the sequential solution.

8.3.3 Experiment 2: N = 50

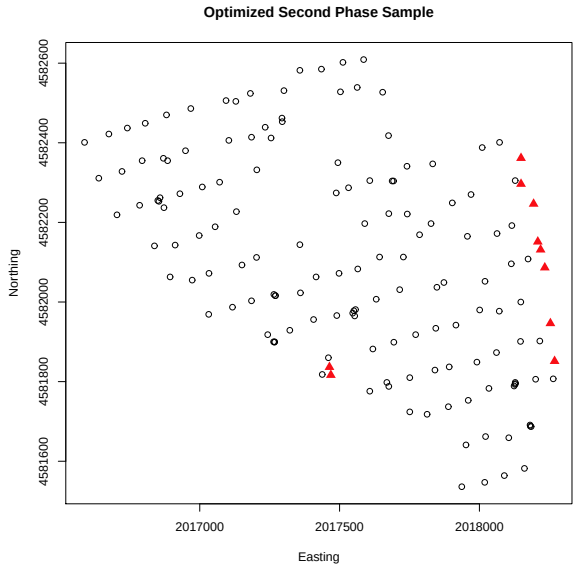
In this final optimization experiment, a full parallel search with 200 different random initial samples of $N = 50$ points is tried. Because it was found that the performance is better with a higher temperature when placing 50 points, for comparability, the starting temperature is set to 2000 in this experiment. Figure 8.19 shows the total distribution of gain for both a starting temperature of 2000 and 1000. The distributions are almost identical. Both acheive a similar distribution of gain overall, with a minimum of 0.20, mean of 0.325, and maximum of 0.5. The best solution with $T = 2000$ is achieves 2.04 and the best with $T = 1000$



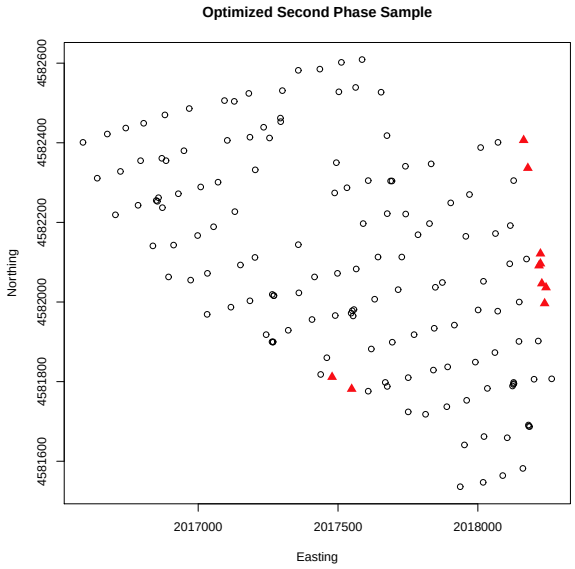
(a) Sequential



(b) Best, Parallel

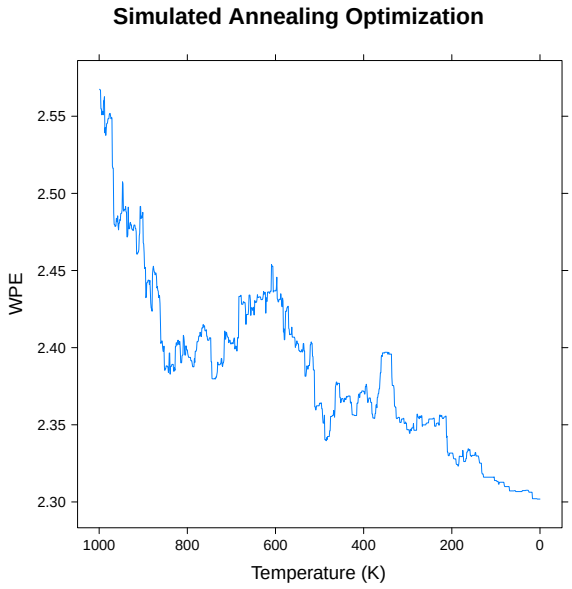


(c) Sequential

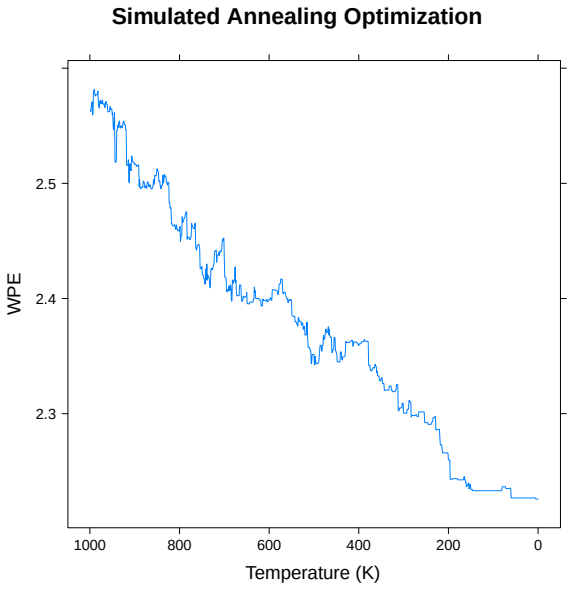


(d) Best, Parallel

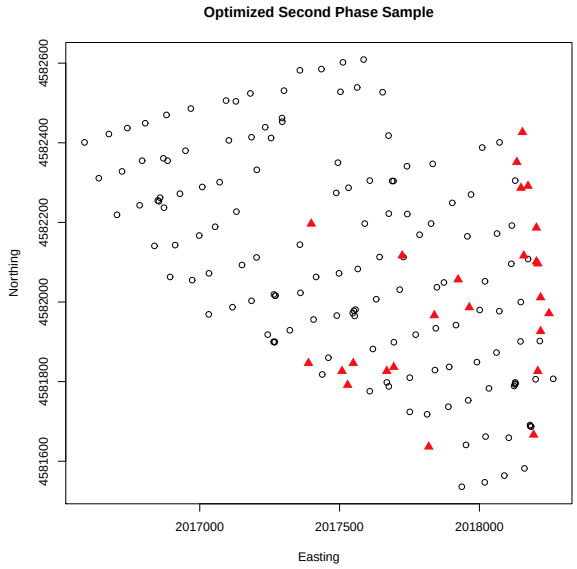
Figure 8.17: Parallel versus sequential performance for $N = 10$.



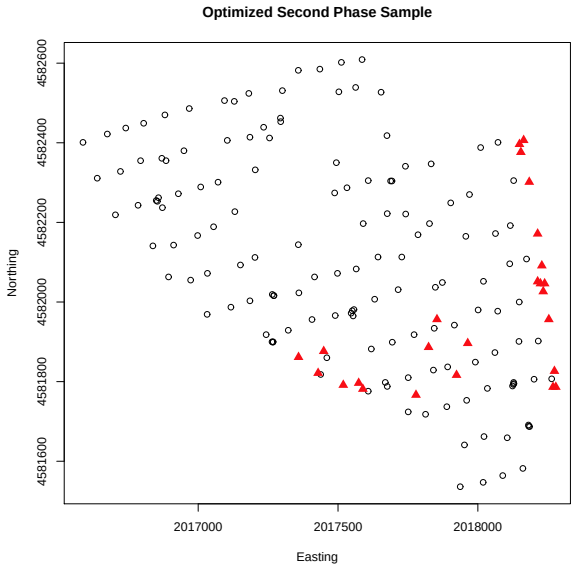
(a) Worst Optimized



(b) Best Optimized



(c) Worst Optimized



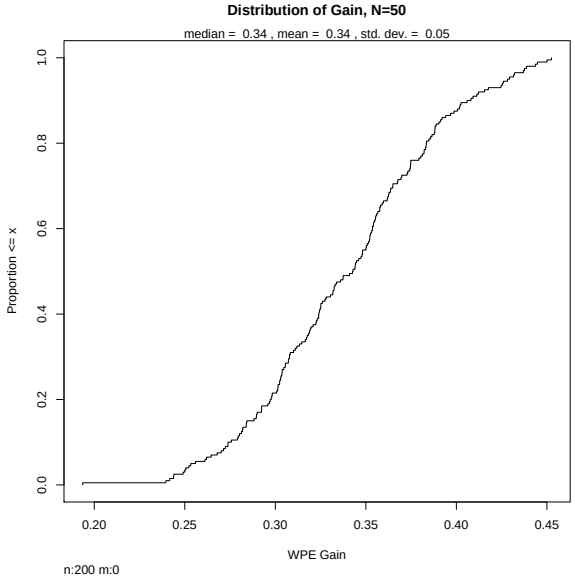
(d) Best Optimized

Figure 8.18: Parallel versus sequential performance for $N = 25$.

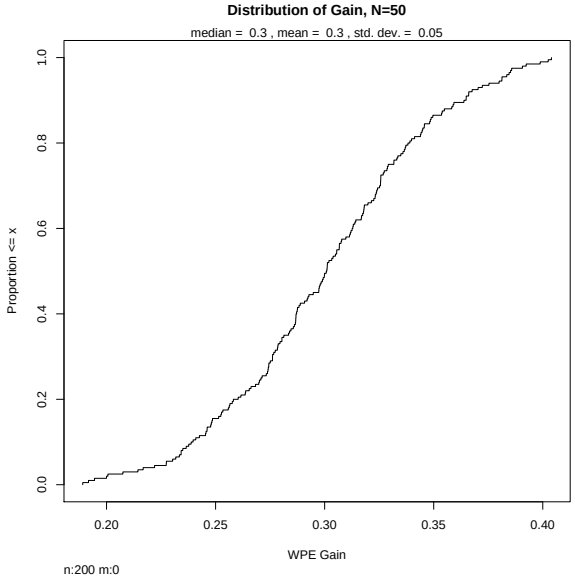
is 2.06, a marginal gain. The difference in the lower end is slightly larger: 2.16 and 2.10 for the worst solutions, respectively. As was seen with $T = 25$ the prior experiment, the sequential simulated annealing run produced a gain of approximately 0.35 (with T=2000), which is the mean of the distribution. Given more random trials, a higher amount of gain, up to 0.5 can be acheived. Qualitatively, however, there is not a substantial difference in the sampling regime suggested by the best parallel solution (shown in figure 8.20d) and the worst (shown in figure 8.20c). Nor is there a substantial qualitative difference between either of these solutions and the sequential solution shown in figure 8.13b. Each of these maps suggests placing measurements in the arc of the main lobe and primary back-lobe of the antenna studied, and at points where there is substantial roughness (i.e., near the transmitter) or large gaps in the measurement lattice. It is interesting that this conclusion is reached, regardless of the approach (or to some extent the thouroughness) of the metaheuristic optimization. From this final experiment, it can be concluded that parallelizing the simulated annealing search does not produce radically different results, although it allows a much greater fraction of the search space to be investigated. For applications where greater certainty is needed with respect to the optimality of the second-phase sample, a massively parallel search may be justified. However, this time may be better spent making a series of phased measurement campaigns with a relatively small N, so that each phase of tuning measurements can build upon the findings of the prior phase. In the next section, these ideas will be tested in a case study of sequential second-phase sampling.

8.4 Case Studies: University of Colorado WiMax

In March 2012, a case study was performed to understand the efficacy and practical gain of the optimized second-phase sampling approach defined above. Although it stands to reason that selecting additional sample points can only improve the realism of the model, and placing them at the places most needed for improvement will improve the model the most, there is still a question of the value of this exercise in practice. On March 29th and 30th, a measurement apparatus identical to the one described in section 6.1.1 was used to collect data at the best sample positions found in the optimizatons described above for $N = 10$, $N = 25$, and $N = 50$. The parallel solutions were used, although as discussed above there does not appear to be a substantive qualitative difference between the parallel and serial solutions. Figure 8.30 shows the

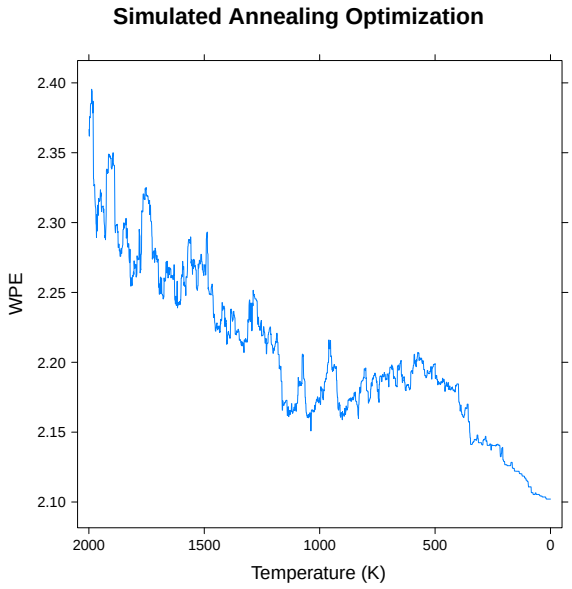


(a) T=2000

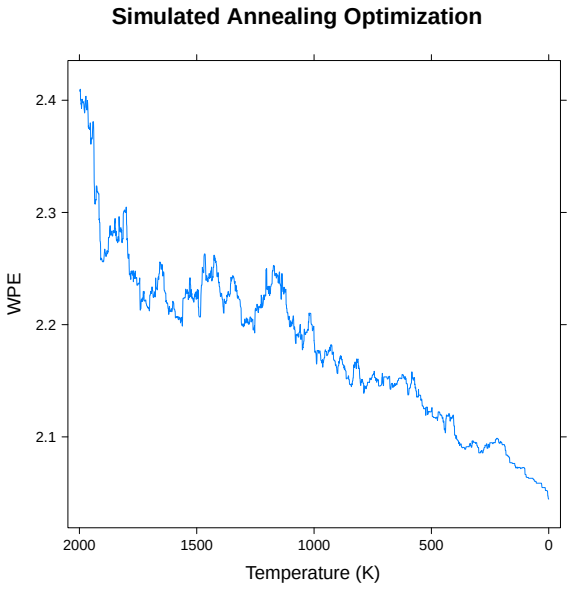


(b) T=1000

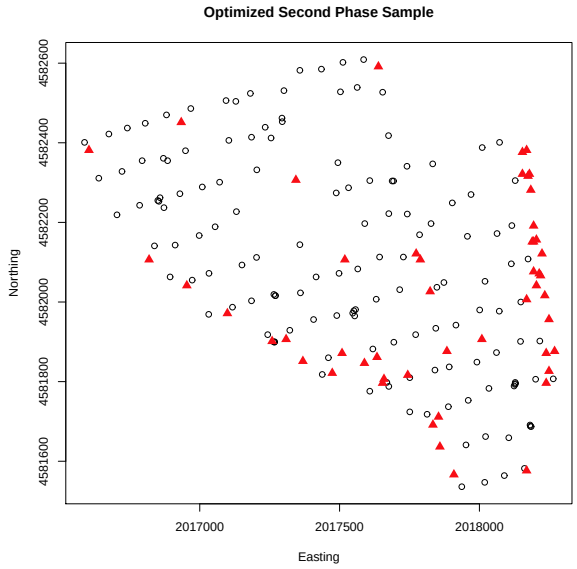
Figure 8.19: Distribution of gain for $N = 50$ with temperatures of 1000 and 2000.



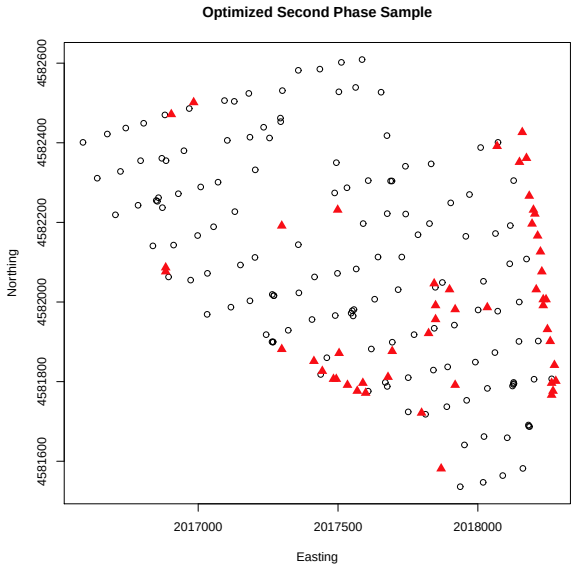
(a) Worst Optimized



(b) Best Optimized



(c) Worst Optimized



(d) Best Optimized

Figure 8.20: Parallel versus sequential performance for $N = 50$.

$N = 50$ sample overlayed on Google Maps orthoimagery. The measurement proceedure used matches that done for the initial sample described in section 6.1: three measurements were taken at each of the 75 points, including physical layer metrics and spectrum flatness. In this test, a WiMax client device (connected to a separate computer) was used to collect simultanous application-layer statistics via a throughput test. Some small overlap exists between the three second-phase sample sets, which allowed some measurements to be used in multiple samples, avoiding collection at 18 duplicate locations.

Figure 8.22 shows the Kriged maps for each second-phase sample as they compare to the original Kriged map. Although the maps look largely similar, there are subtle differences in the shape and size of the dark region, where coverage is poor. Figure 8.23 highlights this distinction by showing the boundary between the covered region (assuming a threshold $CINR \geq 40dB$). Besides producing a qualitatively different and more complete map, a final question is whether the second-phase sampling process has also reduced the residual error of the modeling process. To determine this, two validation methods are used. First, a 10-fold cross validation is done using the sample points themselves. For each fold, a random sample of 20% of the measurement points are predicted using the remainder of the points. Table 8.2 shows the variogram fit statistics and performance results of this test, where the RMSE and MSKV are given as the mean value across the ten folds. There is no clear improvement using these metrics, and in the case of $N = 25$, the error actually increases slightly. It is hypothesized that this is due to the fact that the additional sample locations are a small fraction of the overall measurements and hence, this internal cross validation is insufficiently powered to highlight the differences.

As a more independent performance test, the second metric of improvement involves validation against a random sample of different points. To this end, measurements were made at a random sample of 140 locations within the CU campus boundary, excluding unmeasureable areas per the method described in section 8.2.1. The spectrum analyzer was able to obtain measurements of the cuEN node at 75 points in this random sample. The measurement proceedure was identical to that used in collecting the second-phase samples, with three discrete measurements performed at each point. Then, the interpolated maps are compared to these measurements. As in chapter 7, the performance metrics proposed by Robinson et. al are used for comparability. The results from this experiment are provided in table 8.1. Coverage hole prediction

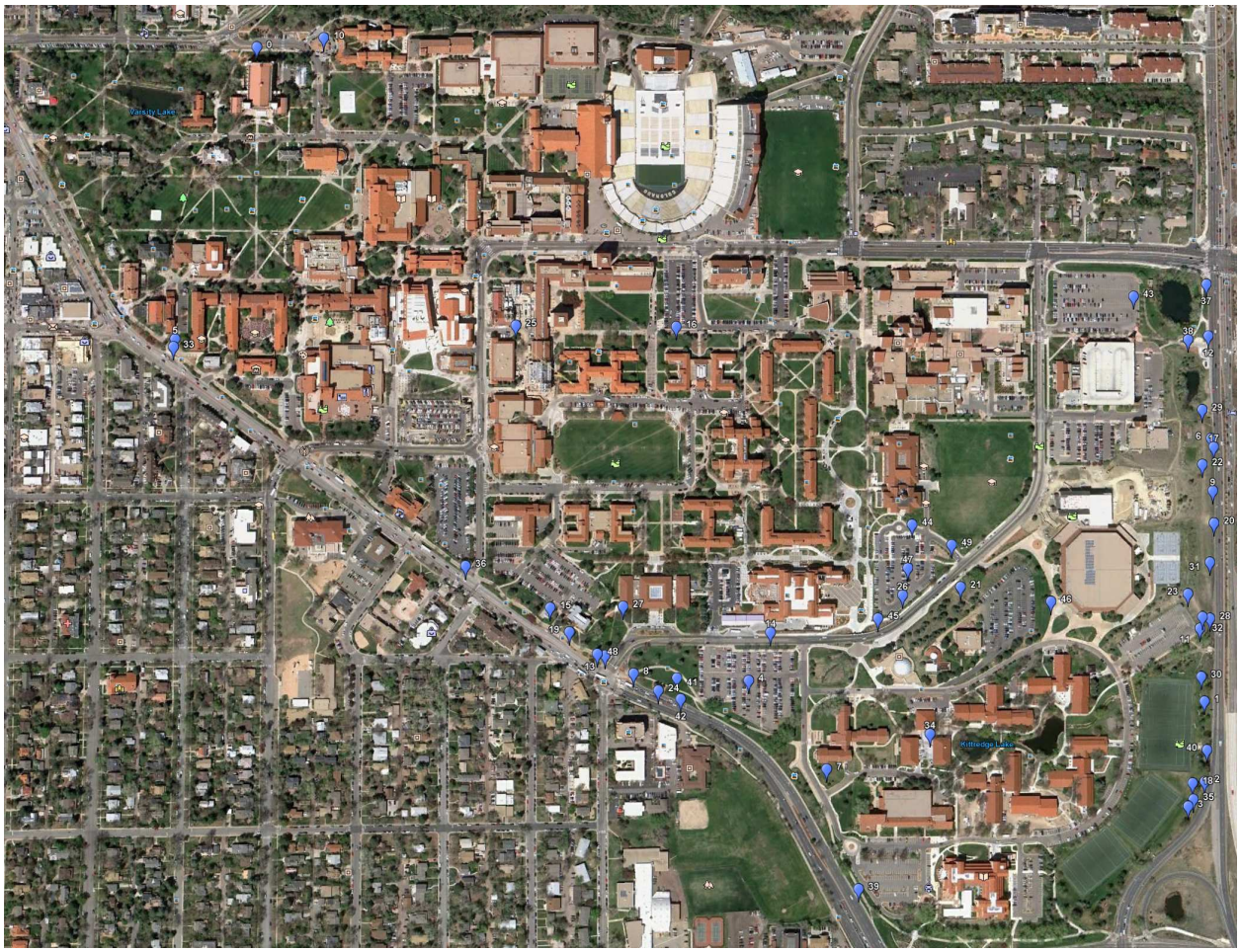


Figure 8.21: Optimized second-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery.

Sample	Hole Finding Accuracy	RMSE
Original	75%	4.07
N=10	71%	4.10
N=25	74%	4.04
N=50	78%	3.95

Table 8.1: Summary of random sample validation and home prediction accuracy for phase 2 samples.

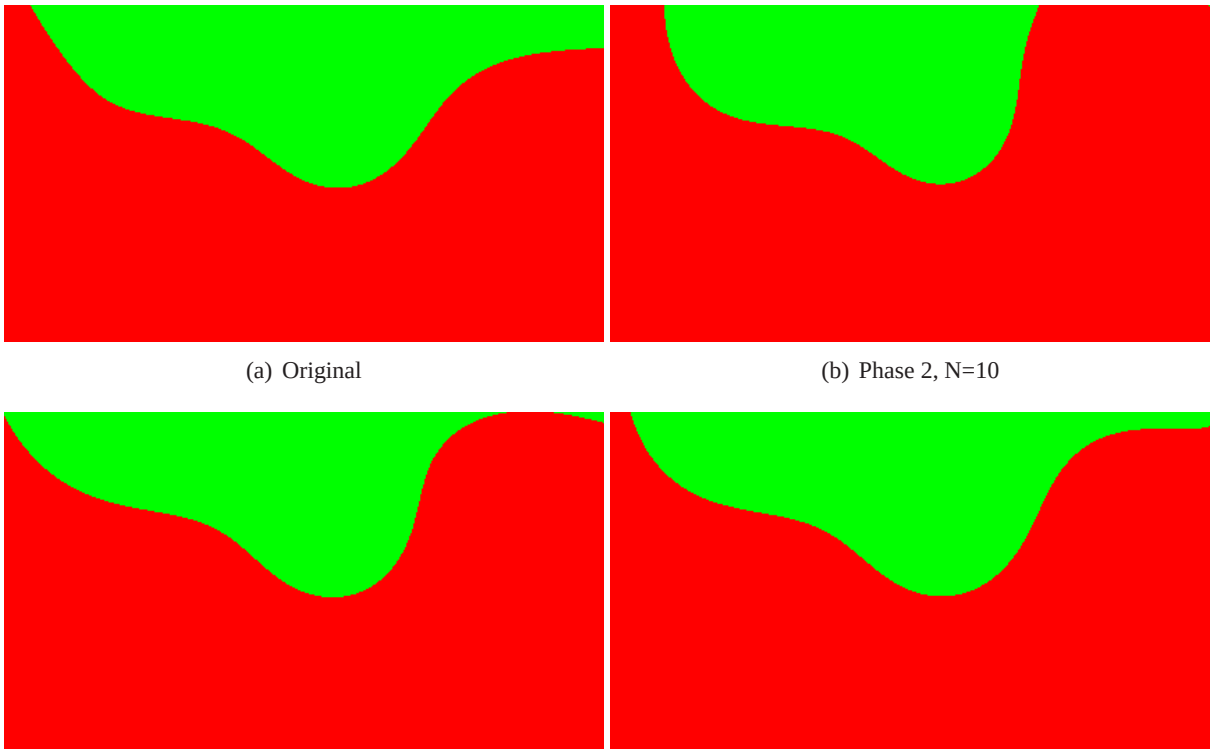
accuracy is largely the same between the maps, with a small improvement for $N = 50$. RMSE is improved only slightly with the second-phase samples.

One interesting observation is that $N = 10$ actually produces a smaller RMSE than $N = 25$. One possible explanation for this fact is related to how second-phase samples are chosen. Optimizing the WPE metric has the effect of choosing locations that (a) have a large Kriging variance, and hence more information is needed at that point and/or (b) have a large "roughness", meaning the map value changes radically near that point. In the smaller samples, the points chosen appear to favor the former reason, placing points near the edges of the map where measurements in the original campaign were sparse. In the $N = 50$ sample, however, there are sufficient phase-two samples that roughness-placed points are also visible in the center of the map. Hence, it may be the case that $N = 10$ capitalizes on the low-hanging fruit of sparse samples and $N = 50$ is able to capitalize on rough areas, but $N = 25$ is the purposeless middle-child, which introduces more variance into the model with additional data, but does not contribute a proportional amount of new and useful modeling information as the other samples do. Based on this observation, it may be the case that the best application of second-phase sampling is either small samples, or large samples, but not medium-sized samples.

As a final comment, it is worth noting something about the sample locations chosen for $N = 50$. Although the choices appear to be relatively random, during data collection it became clear that the loca-

Dataset	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
Original	cubic	1304.05	14.22	20.04	146	TRUE/FALSE	4.00	4.09	12.80
N=10	gaussian	622.89	13.93	22.10	156	TRUE/FALSE	3.90	3.60	12.40
N=25	gaussian	718.31	14.21	27.89	166	TRUE/FALSE	3.92	3.85	12.80
N=50	gaussian	846.24	16.74	28.01	189	FALSE/FALSE	4.21	3.89	11.67

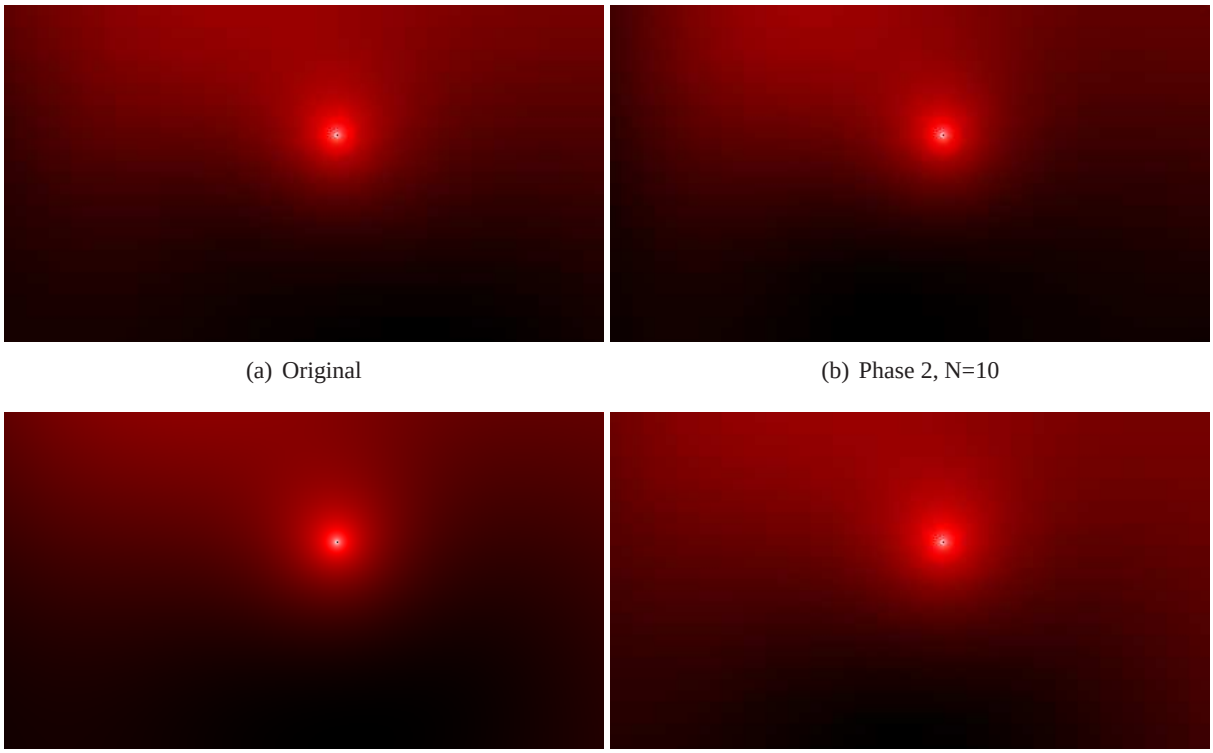
Table 8.2: Summary of cross validation and fit-statistics for phase 2 samples.



(a) Original (b) Phase 2, N=10

(c) Phase 2, N=25 (d) Phase 2, N=50

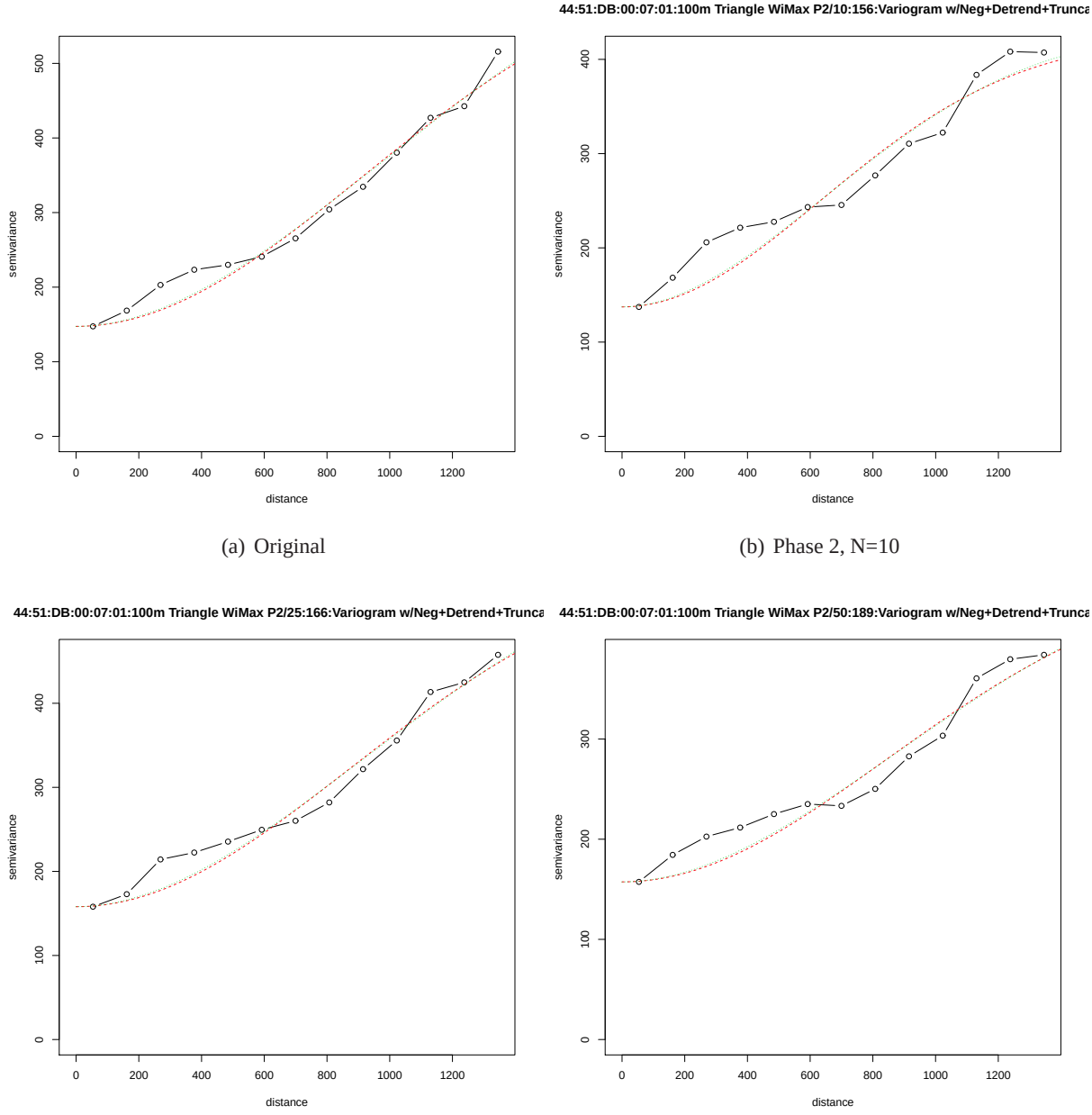
Figure 8.22: Comparison of second-phase threshold maps for CINR=20.



(a) Original (b) Phase 2, N=10

(c) Phase 2, N=25 (d) Phase 2, N=50

Figure 8.23: Comparison of second-phase coverage.



(a) Original

(b) Phase 2, N=10

(c) Phase 2, N=25

(d) Phase 2, N=50

Figure 8.24: Comparison of second-phase variograms.

tions of measurement were actually of particular interest. In fact, many of the measurements were placed at locations in relatively close proximity to the transmitter, and sometimes even with line of sight to the transmitter tower, but with very poor or highly variable observed signal. In this way, the samples for $N = 50$ appear to be grouped around areas where appreciable shadows exist in the RF environment. Because of this phenomenon, it may be possible to identify areas of instability in network coverage simply by producing a large optimized sample and studying where the points are placed. Evaluating this strategy for identifying coverage maladies is an interesting topic for future work.

8.4.1 Excluding Null Measurements

In the previous case study, null measurements were included. Although this provides more information to the optimization process, it was shown in chapter 6 that the best fitting models for this dataset exclude null measurements. To understand how this design decision may affect performance, a second set of measurements was taken at the locations chosen using the same optimization process, but excluding the null measurements. Again, parallel SSA is used with a pool of 200 random initial samples. The resulting best sample and optimization gain plot are given in figure 8.26. This sample is shown overlayed on Google Maps orthoimagry in figure 8.25.

As compared to the first measured second-phase sample, this solution clusters additional points in the northeast corner, and in the parking lot to the north of Folsom field, where measurements were sparse in the original campaign. Figure 8.28 shows the Kriged maps for each second-phase sample, and figure 8.27 shows the boundary between “covered” and “uncovered” points (using the $CINR = 40dB$ threshold derived in the throughput tests described in section 6.1). Inspecting this image makes clear the fact that the optimization process will try to place points at the interface between the covered and uncovered regions, where signal variation (and roughness) is large. Interestingly, this is very similar to the heuristic method used for placing samples suggested in [200], where a push-pull refinement mechanism places samples at the perceived coverage boundary.

As before, two validation methods are used to understand the practical efficacy of this sample. First, a 10-fold cross validation is done using the sample points themselves. For each fold, a random sample of 20%

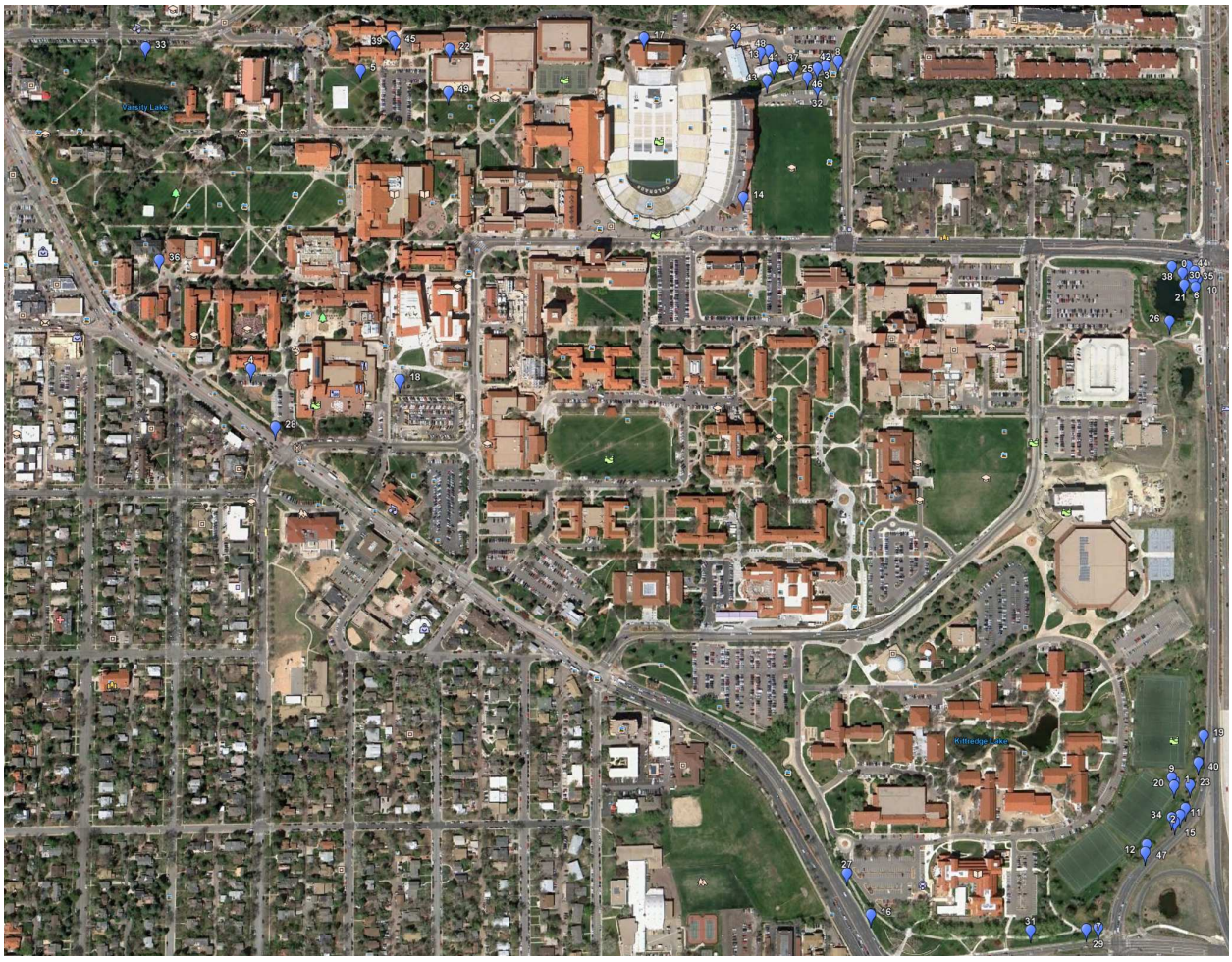
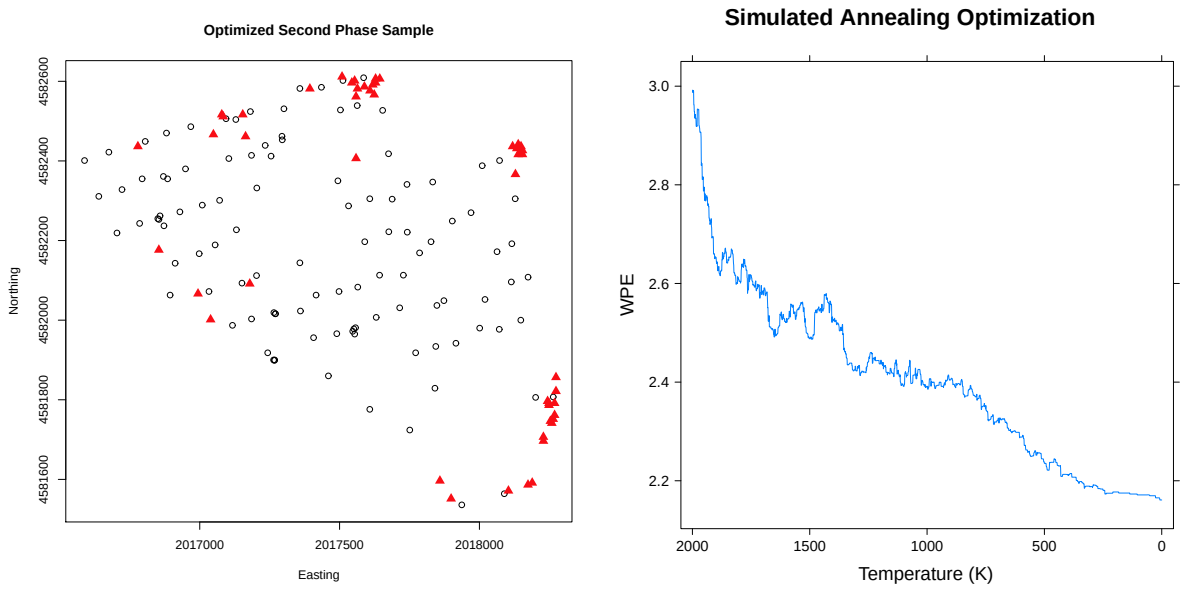


Figure 8.25: Measured second-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery.



(a) Sample

(b) WPE

Figure 8.26: Phase-2 Optimized Sample.

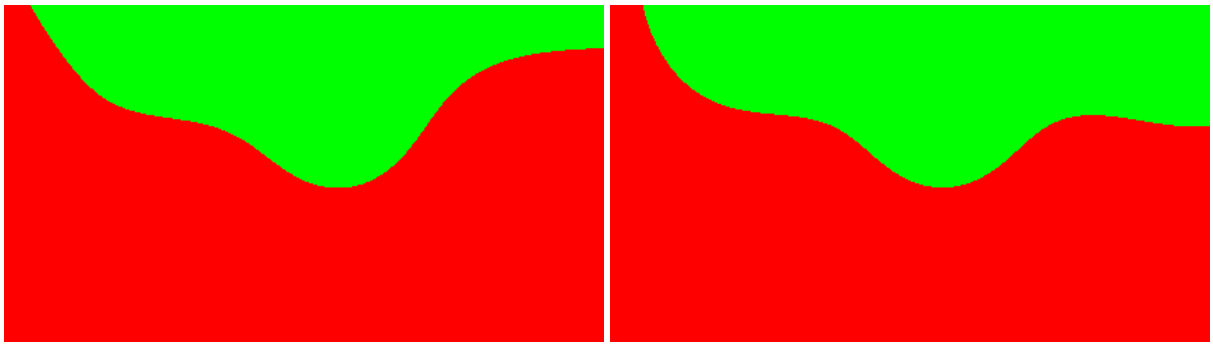
Sample	Hole Prediction Accuracy	RMSE
Original	75%	4.07
N=10	74%	4.04
N=25	74%	4.02
N=50	74%	4.10

Table 8.3: Summary of random sample validation and home prediction accuracy for phase 2 samples.

of the measurement points are predicted using the remainder of the points. Table 8.4 shows the variogram fit statistics and performance results of this test, where the RMSE and MSKV are given as the mean value across the ten folds. There is a clear improvement using these metrics, with each successively large second-phase sample. With $N = 50$, the mean RMSE is actually reduced by 0.8 dB, which is a substantial improvement. The second metric of improvement involves validation against a random sample of different points. To this end, measurements were made at a random sample of 140 locations within the CU campus boundary, excluding unmeasureable areas per the method described in section 8.2.1. The performance with respect to that data set are provided in table 8.3, which shows a marginal improvement in overall RMSE and a small decrease in hole-prediction accuracy. Although the fitted map has a smaller residual error using the second phase sample, this experiment shows that this may not produce a meaningful difference in terms of the practical prediction accuracy of the model. These results are slightly less impressive than those using the null measurements to select points. Although the difference is not large, this suggests that the right optimization strategy may want to include null measurements. Although this is not as effective for fitting this particular data, it may provide useful information about signal boundaries for the optimization process.

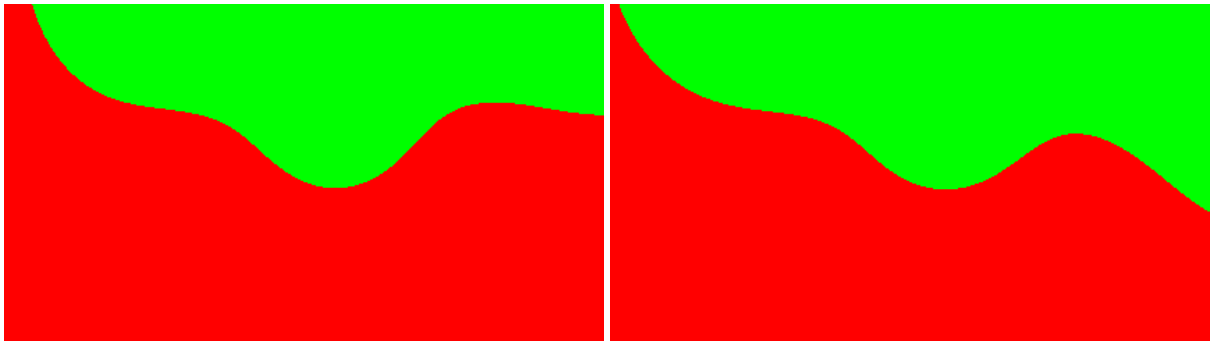
Dataset	Model	ϕ	τ^2	σ^2	N	Trunc/Neg	Mean K-Var	Mean RMSE	Gain
Original	cubic	1304.05	14.22	20.04	146	TRUE/FALSE	4.00	4.09	12.80
N=10	cubic	1768.63	17.09	23.73	152	FALSE/FALSE	4.33	3.82	12.50
N=25	gaussian	620.83	14.37	24.40	168	FALSE/FALSE	3.94	3.85	12.63
N=50	gaussian	372.74	11.83	17.94	194	FALSE/FALSE	3.65	3.29	12.60

Table 8.4: Summary of cross validation and fit-statistics for phase 2 samples.



(a) Original

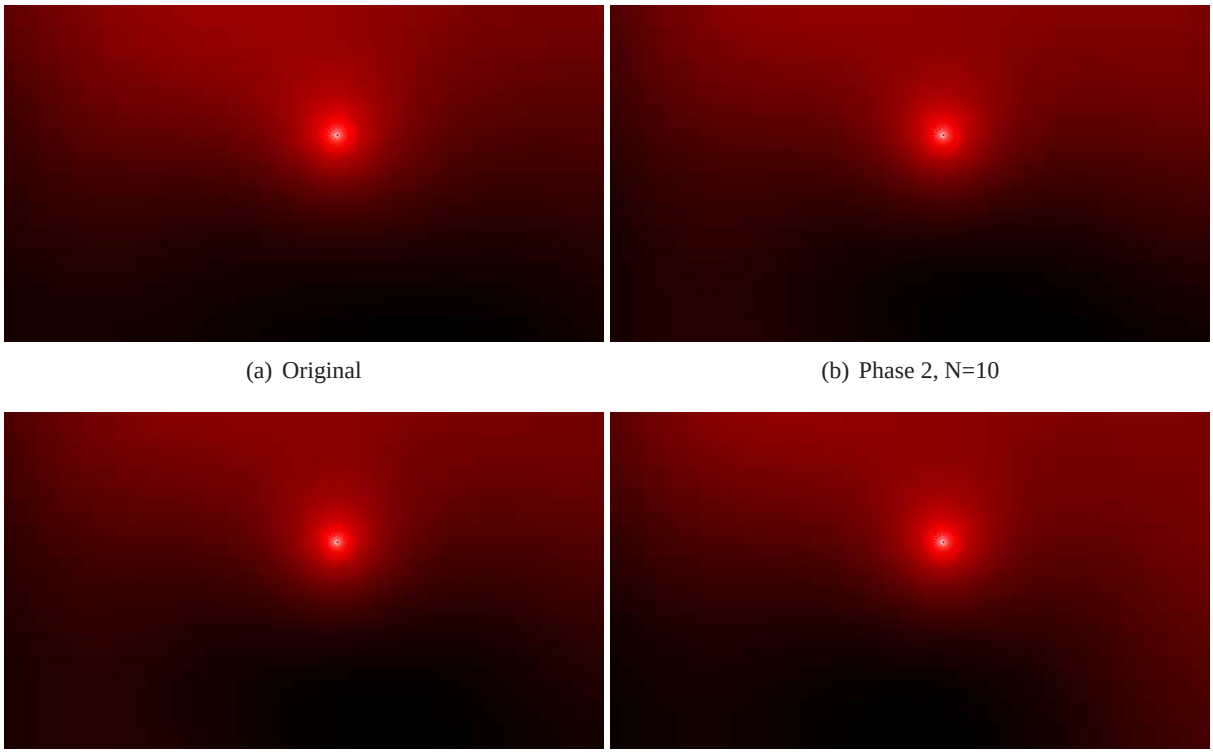
(b) Phase 2, N=10



(c) Phase 2, N=25

(d) Phase 2, N=50

Figure 8.27: Comparison of second-phase threshold maps (threshold is CINR = 40 dB).



(a) Original (b) Phase 2, N=10

(c) Phase 2, N=25 (d) Phase 2, N=50

Figure 8.28: Comparison of second-phase coverage.

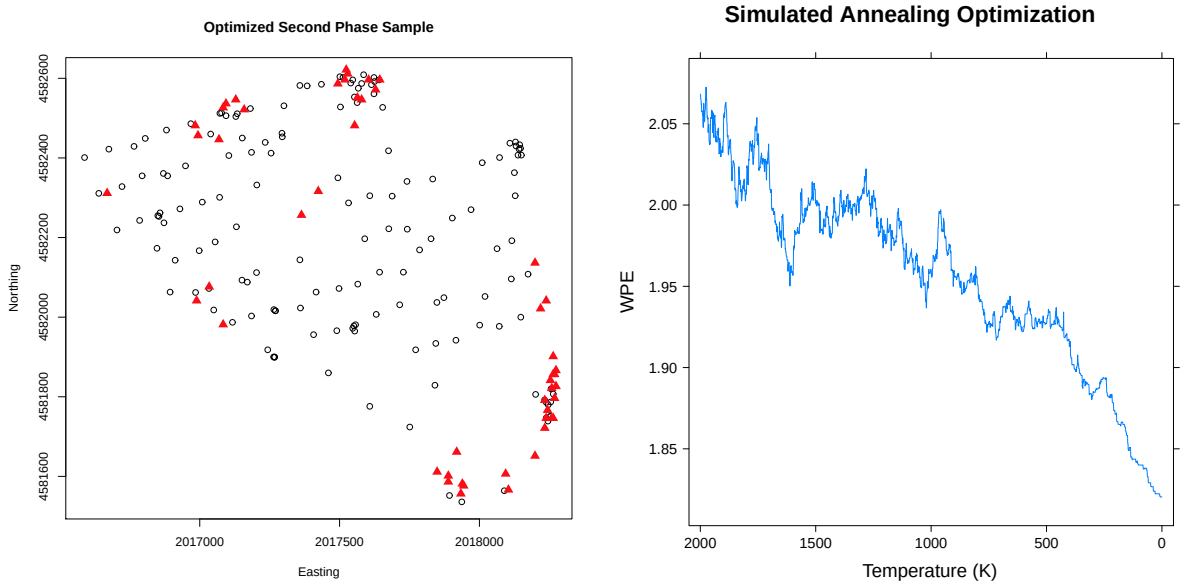
8.4.2 Phase Three Sample

A final research question addressed in this case study is whether additional iterative optimized (i.e., phase three and beyond) samples are worthwhile. To this end, another optimization was performed using the initial and second-phase measurements as input. As before, the best parallel sample was used. Figure 8.29 shows the optimized WPE gain and the resulting sample. At a high level, this sample appears very similar to the phase two sample, with a few important distinctions. First a foremost, there are no measurements clustered in the northeast corner of the map, presumably because the large number of phase-two measurements in that region have sufficiently mapped it. There are also fewer measurements in the northern section of the campus, behind Fulsom field, suggesting that sufficient measurements may have been taken there during the second phase sample. Besides this, the sample is largely similar to the second-phase sample, placing points at regions where prominant shadows exist. According to the optimization, the WPE will be reduced an additional 0.40 over the value obtained after the second-phase sample. However, the important question is whether that maps to a meaningful in terms of improving the predictive value of the map.

Figure 8.31 and figure 8.32 show the resulting maps using the phase-three measurements. By any metric used here these maps are not substantially better than the original map. The cross validation experiment produces a RMSE of 4.19 and MSKV of 4.04, a small increase in RMSE and a small decrease in MSKV over the map produced with the original sample. Using the random sample as a basis for comparison, the hole prediction accuracy and is slightly reduced to 74% and the overall RMSE improved slightly to 4.02, which while better than that of the original map, is no better than the value obtained with a single second-phase sample described in the first case study.

8.5 Summary and Conclusion

This chapter described a new, automatic and intelligent method to select additional coverage mapping sampling locations through optimization. Although this approach has been proposed to some extent in geological and ecological disciplines, it has not previously been applied in computer science, or to the problem of wireless coverage mapping. The chief findings here support the claim that geostatistical multi-



(a) Sample

(b) WPE

Figure 8.29: Phase-3 Optimized Sample.

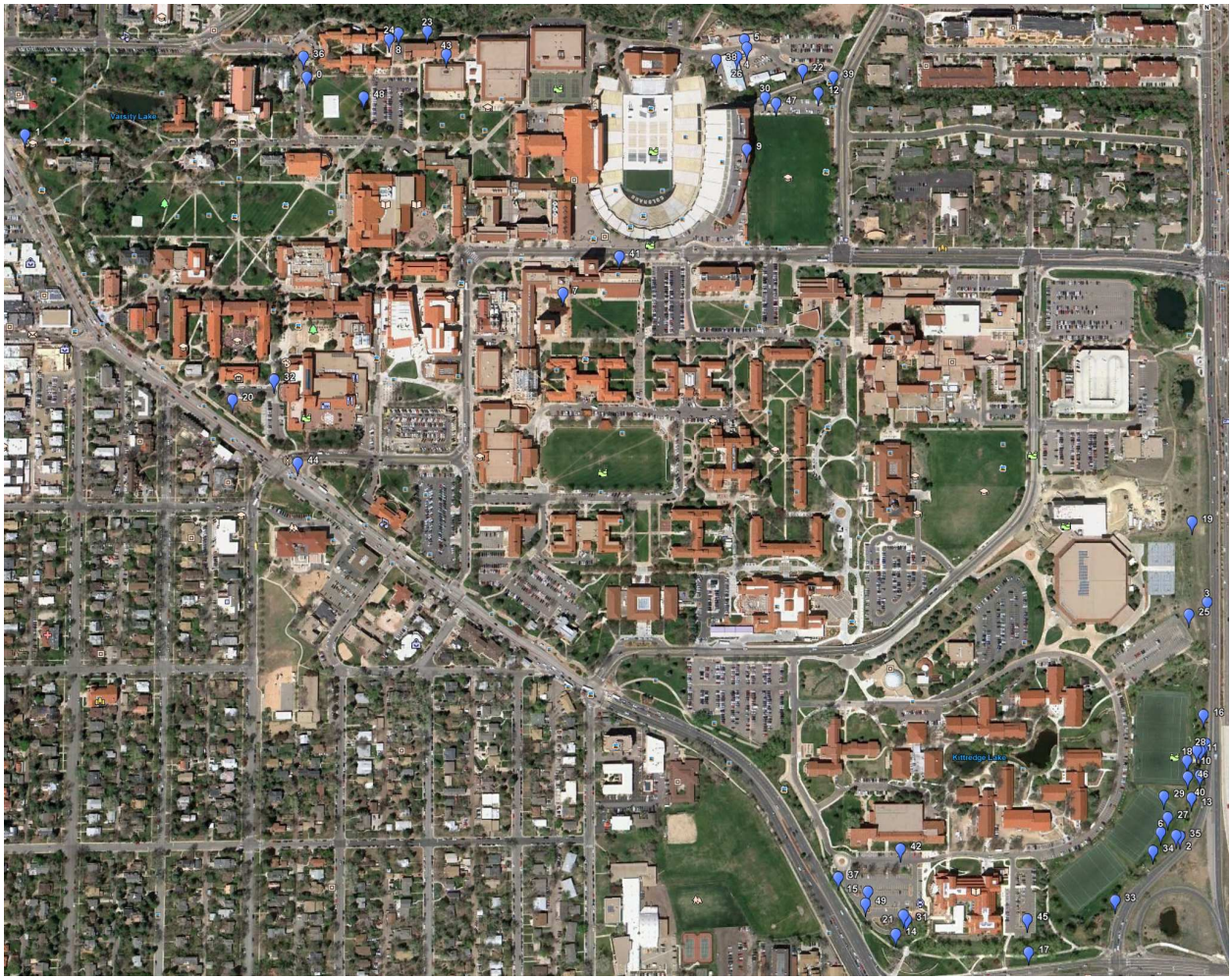
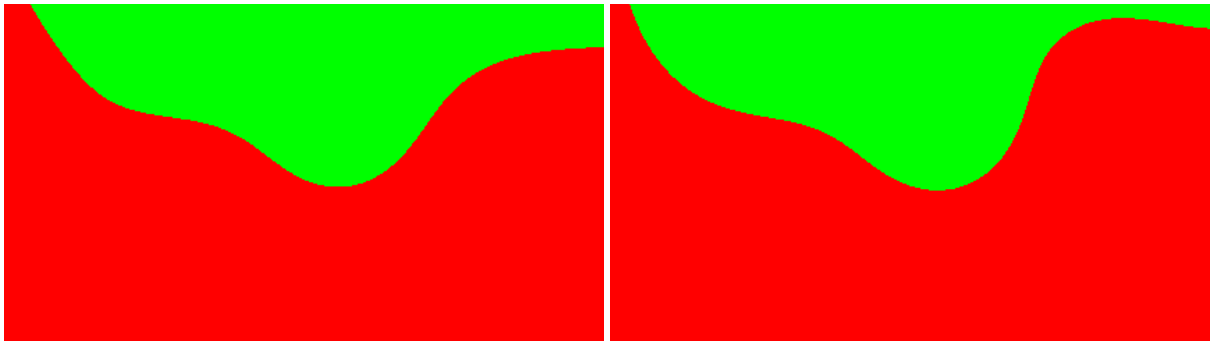
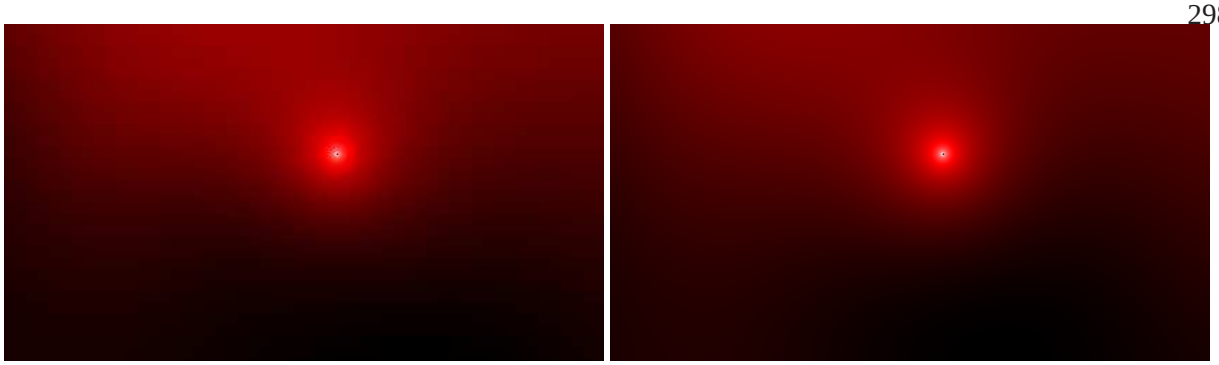


Figure 8.30: Measured third-phase sample for cuEN node with 50 points, overlayed on Google Maps orthoimagery.



(a) Original (b) Phase 3, N=50

Figure 8.31: Comparison of second-phase threshold maps (threshold is CINR = 40 dB).



(a) Original (b) Phase 3, N=50

Figure 8.32: Comparison of second-phase coverage.

phase sample optimization is a reasonable approach to data-driven map refining. Indeed, using this method, fine-tuning a coverage map is a simple matter of “spade work”, where a succession of optimized sampling phases can refine a map as much as is required by the user. Of course, there is an intrinsic lower bound to the accuracy achievable, which is a simple function of the intrinsic variability of the radio environment. However, up to the point where over-fitting can occur, additional samples will only improve the model.

While investigating optimization strategies, myopic (greedy) and metaheuristic approaches were investigated in order to cope with the massive search spaces involved. It was found that greedy approaches perform poorly, but that metaheuristic approaches such as simulated annealing perform very well and in a reasonable amount of time. Although attractive at first, large-scale parallel optimization does not appear to provide substantive improvement over serial optimization. At a high level, these results seem to support the approach to learning systems and data collection taken by active learning systems in general, and hopefully these results will motivate additional work on approaches to learning and model refining that take into account the careful selection of measurements when refining their fits.

To understand the practical value of this approach, a case study was performed focusing on improving the coverage map of the cuEN WiMax BS on the University of Colorado Campus. In this case the iterative optimized sampling was able to produce only small gains in the fidelity of the resulting map. It may be the case that the improvement is marginal because the original sample is sufficient for this particular network and additional samples have little to gain over the intrinsic variation of the channel (which appears to be

approximately 3-4 dB for this network). However, the optimized sampling process itself was able to provide insight into the network coverage not readily available in the coverage maps; samples were located at areas of large variation and near the boundaries of practical network coverage.

Although this chapter makes an important first step in the direction of applying optimized sampling to the RF environment, there are several open questions that deserve to be addressed before it sees widespread use. In particular, it may be the case that optimized sampling produces larger gains in environments where the original sample is less principled (i.e., sparse, biased, or irregular). In this way, the optimized sampling process may be best suited for addressing deficiencies in the initial sample. Similarly, it may be worthwhile to specifically direct the optimization process towards particular areas of interest, so that optimal samples can focus on domain-specific needs in addition to reducing variance. Understanding the answers to these questions, as well as applying the sample optimization process to additional networks and environments to understand where it excels and where it is unnessicary is an important area for future investigation.

Chapter 9

Conclusion

This thesis began with the claim that *there are not good methods for determining how well a given network works over a given area and presenting this information in a meaninful way*. As a possible solution to this problem, the application of geostatistical mapping methods were proposed, adapting mathematics developed for geological mining applications to a new and vastly different domain. Ultimately, it was found that this is a reasonable application, and the robust spatial statistical methods used in geostatistics allow for the creation of coverage maps that embrace, rather than ignore, the spatiotemporal variability of the wireless channel. In the case studies presented above, geostatistical approaches were shown to produce maps with a fine accuracy and much better predictive performance than standard *a priori* models that do not use measurements, or simple measurement-based fitting. However, to focus only on the performance improvement is to miss the real value of the geostatisitcal methods: by implementing an appropriate sampling design, modeling the underlying spatial structure of the data, and using a statistical method, an interpolated map can be generated with a well defined notion of residual error: the prediction at each point is a distribution, not simply a value. Additionally, this robust coverage map can be produced using a reasonably small amount of easily obtained data (several hundred samples for a space the size of a large university campus), which amounts to a tractable amount of routine "spade work" (approximately three days work for a single dedicated experimenter).

In order to enable these results, new mechanisms for measurement were developed and paired with statistically safe sampling methodologies and interpolation techniques. In addition, careful attention was paid to the comparative value of performance metrics, so that the resulting maps are not only well-fitting

to the data, but also communicate meaningful information about the real performance, and underlying variability of that performance, at the interpolated locations.

In addition to this core work, several important tangential threads were investigated. In particular, several extensions were evaluated that provide features that would be useful were the methods proposed here to be widely adopted. First, the prospect of resampling was investigated to understand how a coverage map might be derived from measurements collected at locations where it is convenient to collect, but with some substantial sampling bias (for instance, in city streets). It was found that resampling of this data can help to alleviate bias and that the resampled data can be well-modeled with geostatistical techniques. In fact, the resulting coverage maps are as accurate at predicting coverage holes as state-of-the-art iterative heuristic refinement methods (e.g., [200]), with a nearly identical number of measurements. This is an exciting result because this performance is obtained with the same amount of effort, while producing a substantially richer coverage map, where each interpolated point is a value distribution instead of a binary value. Next, the prospect of crowd-sourced coverage mapping was investigated, where many volunteers might cooperate to collect the measurements for a coverage map. It was found that this may be a feasible approach to coverage mapping, if a sufficient fraction of the population inhabiting the mapped location is willing to participate. However, a case study using data collected with a production crowd-sourcing system showed that in practice this level of participation may not yet be present. Finally, as a way of refining and tuning the generated coverage maps, sample optimization was proposed and investigated. It was found that metaheuristic approaches to sample optimization perform well, and that an insightful second-phase sample can be found in several hours computation on a single computer. These additional optimized samples provide valuable insight into the coverage of a given transmitter by identifying areas where variance is high, near coverage thresholds for the network. Although quantitative improvements in predictive performance were shown to be small in a case study, this iterative optimized sampling strategy shows promise in this domain and deserves further investigation.

As is typical of the scientific process, this work has also brought to light a number of areas where future work is needed:

- Better methods are needed for visualization of wireless coverage maps. The color mapping and interactive map overlaying described in section 5.4.7 are only a beginning in terms of visualization strategies. In particular new methods are needed to draw out contours and highlight holes and deviations. Mapping systems that perform dimension reduction to simultaneously communicate value and variance are most needed. One can imagine a network planning tool that provides for interactive mapping and surveying of a network region, while interleaving GIS sources and orthoimagery smoothly. The method presented here could be easily integrated into such a system and the map data adapted to any such visualization method, however substantial further work is needed in order to understand which visualization strategies work best, and in which situations.

- Accurate, fast, and inexpensive measurement hardware are needed. The spectrum analyzer, drive-test software, and UE radio devices used in this study all presented substantial shortcomings. The most accurate tools were also very slow, cumbersome, and expensive. Meanwhile, COTSE-based devices are faster, but they provide fewer options for measurement, are generally closed to low-level driver modification and analysis, and produce noisy results that prove difficult to fit. Developing better mobile measurement hardware that is open and modifiable, easy to use both by experts and technicians, and provides accurate and useful metrics, would be a huge boon to the coverage mapping problem. One can imagine a “smart” measurement device that collects and actively guides the measurement process using the methods described here. Developing similar sensor systems for long-term spectrum sensing and mapping deployment would also have tremendous value as cognitive and whitespaces networking gains traction.

- The work in section 7.2 on crowd-sourced coverage mapping demonstrations that this area is ripe for further work, as evidenced by the fact that companies like OSM have already deployed software to collect this data with smart phones. However, it is still not clear the best way to cope with the relative sparsity and noise that is an intrinsic component of crowd-sourced data collection. Substantial work is needed to understand the practical accuracy of measurement using common mobile UE hardware, how to collect sufficient data without effecting the battery life of small mobile

devices, and whether a sufficient fraction of users would be available to collect data in the regions it is needed most. Successful crowd-sourcing data collection projects like the “Test my ISP” project by the FCC are encouraging [50].

- The geostatistical coverage mapping method developed in this thesis makes use of standard Kriging approaches, like OK and omnidirectional models. However, more advanced Kriging methods might make way for further modeling gains. For instance, anisotropic models may offer a way to more finely model the coverage of transmitters with directional antennas. These models segment the azimuthal plane and fit each segment with a possibly different geostatistical model (and/or method). Higher-powered Kriging approaches, such as UK, which allow for the mean of the map to be modeled by an arbitrary function, or local Kriging which more carefully considers the “neighborhood” of measurements around a given pixel when fitting it, may offer additional gains. However, initial experimentation in this direction suggested that these approaches may also be needlessly over-powered (or myopic in the case of local Kriging) for the application to coverage mapping.

- In this work it was assumed that all areas within the region of interest are equally valuable to map. In practice, this is seldom the case; typically some areas receive more use or are more important for mapping than others. Identifying domain-appropriate stratified sampling designs, which sample some areas more densely than others is an interesting topic for future work. A similar approach could be used to guide multi-phase sample optimization, extending the methods proposed in chapter 8.

- Section 3.7 provided an analysis of the accuracy of commerical raytracing systems when predicting the propagation in one environment. Although the initial goal of this experiment was to determine the relationship between input data fidelity and raytracing prediction accuracy, it was found that these systems are not able to deal with high-resolution environmental data. And, even moderately complex data may lead to issues with computing knife-edge diffractions, leading to compounding errors. In general, better raytracing algorithms are needed that can make use of high resolution environment data, be it created through crowd-sourcing using design tools like Google SketchUp

[14], or painstakingly collected using arial scanning LiDaR. Being able to utilize this data will require substantial advances in terms of meaningful vectorization of point data, preprocessing, and UTD computation, perhaps utilizing substantial parallelism. And, once such a system is developed, serious work is needed in terms of verifying the accuracy of these ray-tracing systems and understanding the fundamental relationship this accuracy has with the fidelity of input data.

- Although some work was done here to parallelize geostatistical computation (e.g., the parallel Kriging variance implementation in section C.1), these methods are still quite computationally complex. Some trivial parallelization is possible, by simultanously mapping measurements from multiple APs, but the underlying functions, particularly those involving large matrix operations, could still be optimized and parallelized substantially. In order for geostatistical mapping methods of the sort proposed here to be widely integrated into desktop planning software, or even a handheld measurement device, some work will be needed to do finer parallelization on, e.g., General Purpose Graphical Processing Unit (GPGPU) hardware.

- Finally, The methods described here have been limited in their analysis to outdoor microcell networks operating in the Ultra High Frequency (UHF) band. Determining the efficacy of the methods when applied to other types of networks at different frequencies is an important area for further work. These methods could also be trivially adapted to indoor or three-dimensional mapping (e.g., in multi-floor buildings or Unmanned Aerial Vehicle (UAV) applications), however substantial work will be needed to determine their practical accuracy in these settings and whether domain-specific modifications will be necessary.

In sum, this thesis has provided a complete and functional system for mapping the coverage of a production wireless network. Although the results here cannot be extrapolated to any networking techology in any environment, these results appear to hold promise for the broad application of geostatistical mapping to the RF environment. All told, the future appears bountiful for additional work in this area. It is the humble hope of this thesis that the work done here will help enable exciting new technological solutions to the problems faced with wireless networks; in effect, helping to transform them from an amazing technology

that “seldom works as well as one would like”, to a technology that is at the same time reliable, ubiquitous, and essential.

Bibliography

[1] Radiocommunications regulations (general user radio licence for fixed radio link devices) notice 2004. http://www.rsm.govt.nz/cms/licensees/types-of-licence/general-user-licences/fixed-radio-link-devices/. Pursuant to Regulation 9 of the Radiocommunications Regulations 2001 made under section 116 (1) (b) of the Radiocommunications Act 1989.

[2] Ersi shapefile technical description. Whitepaper, Environmental Systems Research Institute, Inc., July 1998.

[3] Discussion with SkyPilot engineers. Personal Communication, June 2007.

[4] Uptown Services. http://www.uptownservices.com/, March 2008.

[5] Lte; evolved universal terrestrial radio access (e-utra); physical layer proceedures, October 2009.

[6] CRAWDAD: A Community Resource for Archiving Wireless Data At Dartmouth. `http://crawdad.org/`, October 2010.

[7] MPC. http://www.multiprecision.org/, June 2010.

[8] Radiotap. http://www.radiotap.org/, November 2010.

[9] Rural Link. http://www.rurallink.co.nz/, November 2010.

[10] The GNU MPFR Library. http://www.mpfr.org/, June 2010.

[11] The GNU Multiple Precision Arithmetic Library. http://gmplib.org/, June 2010.

[12] The MadWiFi Project. http://madwifi-project.org/, November 2010.

[13] University of Waikato WAND Network Research Group. http://wand.net.nz, November 2010.

[14] Google SketchUp. http://sketchup.google.com/, January 2011.

[15] Kismet. http://www.kismetwireless.net/, March 2011.

[16] Netstumbler. http://www.netstumbler.com/, March 2011.

[17] Openwrt. http://openwrt.org/, March 2011.

[18] Some commonly asked questions about emc measurement units. http://www.microvolt.com/table.html, January 2011.

[19] V.S. Abhayawardhana, I.J. Wassell, D. Crosby, M.P. Sellars, and M.G. Brown. Comparison of empirical propagation path loss models for fixed wireless access systems. In VTC 2005-Spring, volume 1, pages 73–77, May 2005.

[20] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level measurements from an 802.11b mesh network. SIGCOMM Computer Communications Review, 34(4):121–132, 2004.

[21] K. Allsebrook and J.D. Parsons. Mobile radio propagation in brittish cities at frequencies in the vhf and uhf bands. IEEE Transactions on Antennas and Propagation, VT-26:313–323, 1977.

[22] Mohammad H. Firooz anda Dustin Maas, Junxing Zhang, Neal Patwari, and Sneha K. Kasera. Channel sounding for the masses: Low complexity gnu 802.11b channel impulse response estimation. Technical report, University of Utah: Sensing and Processing Networks (SPAN), 2010.

[23] Christopher R. Anderson and Charles B. Cameron. Weak-signal spectrum survey measurements and analysis for cognitive radio. In Proceedings of the SDR'08 Technical Conference and Product Exposition, 2008.

[24] C.R. Anderson and T.S. Rappaport. In-building wideband partition loss measurements at 2.5 and 60 ghz. Wireless Communications, IEEE Transactions on, 3(3):922 – 928, May 2004.

[25] Eric Anderson, Caleb Phillips, Kevin Bauer, Douglas Sicker, and Dirk Grunwald. Modeling directionality in wireless networks [extended abstract]. In ACM SigMetrics, 2008.

[26] Eric Anderson, Caleb Phillips, Douglas Sicker, , and Dirk Grunwald. Modeling environmental effects on directionality. Mathematical and Computer Modeling Journal: Special Issue on Modeling and Simulation of Wireless Networks, 2010. (to appear).

[27] Eric Anderson, Caleb Phillips, Douglas Sicker, and Dirk Grunwald. Modeling environmental effects on directionality in wireless networks. Technical report, University of Colorado at Boulder, 2008.

[28] Eric Anderson, Caleb Phillips, Douglas Sicker, and Dirk Grunwald. Modeling environmental effects on directionality in wireless networks. In 5th International Workshop on Wireless Network Measurements (WiNMee), 2009.

[29] Eric Anderson, Caleb Phillips, Gary Yee, Douglas Sicker, and Dirk Grunwald. Challenges in deploying steerable wireless testbeds. In TridentCom 2010, May 18-20 2010.

[30] Eric Anderson, Gary Yee, Caleb Phillips, Douglas Sicker, and Dirk Grunwald. The impact of directional antenna models on simulation accuracy. In 7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2009.

[31] AT&T. AT&T Coverage Viewer. http://www.wireless.att.com/coverageviewer/, July 2011.

[32] M. Emin Aydin and Vecihi Yigit. Parallel Metaheuristics. Wiley Series on Parallel and Distributed Computing. Wiley Interscience, 2005.

[33] F. Babich, M. Comisso, M. D'Orlando, and L. Manià. Interference mitigation on wlans using smart antennas. Wireless Personal Communications, 36(4):387–401, 2006.

[34] Les Barclay. Propagation of radiowaves. Institution of Electrical Engineers, 2003.

[35] Sam Bartels and Murray Pearson. Wireless local area network planning: An overview. In New Zealand Computer Science Research Student Conference, 2008.

[36] D. R. Bellhouse. Optimal designs for sampling in two dimensions. Biometrika, 64(3):605–611, December 1977.

[37] A. Blomquist and L. Ladell. Prediction and calculation of transmission loss in different types of terrain. NATA AGARD Conf. Pub. CP. 144, Res. Inst. Nat. Defense, Dept. 3, S-10450:32/1–32/17, 1977.

[38] David Brown and Gregory M. Stone. A report on technology independent methodology for the modeling, simulation, and empirical verification of wireless communication system performance in noise and interference limited systems operation on frequencies between 30 and 1500 mhz. Technical report, TIA, May 1997.

[39] Whitney Bryen. Cu-boulder offers seamless wi-fi on buff bus. Daily Camera, 17, August 2011.

[40] M.C. Bueso, J.M. Angulo, and F.J. Alonso. A state-space model approach to optimum spatial sampling design based on entropy. Environmental and Ecological Statistics, 5:29–44, 1998.

[41] Michael Buettner, Eric Anderson, Gary Yee, Dola Saha, Doug Sicker, and Dirk Grunwald. A phased array antenna testbed for evaluating directionality in wireless networks. In MobiEval '07, San Juan, Puerto Rico, USA, June 2007. ACM.

[42] K. Bullington. Radio propagation for vehicular communication. IEEE Trans on Ant and Prop, VT-26:295–308, 1977.

[43] Simon Byers and Dave Kormann. 802.11b access point mapping. Communications of the ACM, 46(5):41–46, 2003.

[44] Joseph Camp, Joshua Robinson, Christopher Steger, and Edward Knightly. Measurement driven deployment of a two-tier urban mesh access network. In Proceedings of the 4th international conference on Mobile systems, applications and services, MobiSys '06, pages 96–109, New York, NY, USA, 2006. ACM.

[45] Kin Lien Chee, S.A. Torrico, and T. Kurner. Foliage attenuation over mixed terrains in rural areas for broadband wireless access at 3.5 ghz. Antennas and Propagation, IEEE Transactions on, 59(7):2698 –2706, july 2011.

[46] Jean-Pail Chilés and Pierre Delfiner. Geostatistics: Modeling Spatial Uncertainty. Wiley, 1999.

[47] Steve Chilton. Crowdsourcing is radically changing the geodata landscape: case study of openstreetmap. In Proceedings of the UK 24th International Cartography, 2009.

[48] Dieter J. Cichon and Thomas Kürner. Digital mobile radio towards future generation systems: Cost 231 final report. Technical report, COST European Cooperation in the Field of Scientific and Technical Research - Action 231, 1993.

[49] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active learning with statistical models. Journal of Artificial Intelligence Research, 4:129–145, 1996.

[50] Federal Comunications Commision and Sam Knows. Test my isp. http://www.testmyisp.com/, April 2011.

[51] AWE Communications. Wave propagation and radio network planning. http://www.awe-communications.com/, January 2012.

[52] K. Connelly, Yong Liu, D. Bulwinkle, A. Miller, and I. Bobbitt. A toolkit for automatically constructing outdoor radio maps. In Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05), volume 2, pages 248 – 253 Vol. 2, apr. 2005.

[53] JDS Uniphase Corporation. E6474a wireless network optimization software. http://www.jdsu.com/en-us/Test-and-Measurement/Products/a-z-product-list/Pages/e6474a-wireless-network-optimization-software.aspx, September 2011.

[54] Roger Coudé. Radio Mobile. http://www.cplus.org/rmw/english1.html, July 2010.

[55] Noel A. C. Cressie. Statistics for Spatial Data. Wiley, revised edition, 1993.

[56] Emiliano Dall'Anese. Geostatistics-inspired sparsity-aware cooperative spectrum sensing for cognitive radio networks. In Proceedings of the Second International Workshop on Mobile Opportunistic Networking, MobiOpp '10, pages 203–204, New York, NY, USA, 2010. ACM.

[57] Jack Damelin, William A. Daniel, Harry Fine, and George V. Waldo. Development of vhf and uhf propagation curves for tv and fm broadcasting. Technical Report R-6602, Federal Communications Comission (FCC), September 1966.

[58] William Daniel and Harry Wong. Propagation in suburban areas at distances less than ten miles. Technical report, Federal Communications Commission, Office of Engineering and Technology, 1991.

[59] Y.L.C. de Jong, M.H.J.L. Koelen, and M.H.A.J. Herben. A building-transmission model for improved propagation prediction in urban microcells. IEEE Transactions on Antennas and Propagation, 53(2):490 – 502, march 2004.

[60] J.N.C. de Oliveira, M.S. Alencar, and V.C. da Rocha. A new propagation model for cellular planning. In IEEE International Telecommunications Symposium, 2006.

[61] Rafael S. de Souza and Rafael D. Lins. A new propagation model for 2.4 ghz wireless lan. In APCC, 2008.

[62] Gilles Y. Delisle, Jean-Paul Lefévre, Michel Lecours, and Jean-Yves Chouinard. Propagation loss prediction: A comparative study with application to the mobile radio channel. IEEE Transactions on Antennas and Propagation, VT-34:86–96, 1985.

[63] Eric M. Delmelle and Pierre Goovaerts. Second-phase sampling designs for non-stationary spatial variables. Geoderma, 153:205–216, 2009.

[64] G. Durgin, T.S. Rappaport, and Hao Xu. Measurements and models for radio path loss and penetration loss in and around homes and trees at 5.85 ghz. IEEE Transactions on Communications, 46(11):1484 –1496, nov 1998.

[65] G.D. Durgin. The practical behavior of various edge-diffraction formulas. IEEE Antennas and Propagation Magazine, 51(3):24 –35, 2009.

[66] ECC. The analysis of the coexistence of FWA cells in the 3.4 - 3.8 GHz bands. Technical Report 33, European Conference of Postal and Telecommunications Administrations, 2003.

[67] R. Edwards and J. Durkin. Computer prediction of service areas for v.h.f. mobile radio networks. Electrical Engineers, Proceedings of the Institution of, 116(9):1493 –1500, september 1969.

[68] R.E. Edwards and J. Durkin. Computer prediction of field strength in the planning of radio systems. In Proc. Inst. Elec. Eng., 1969.

[69] John J. Egli. Radio propagation above 40 mc over irregular terrain. In Proceedings of the IRE (IEEE), volume 45, pages 1383–1391, October 1957.

[70] Shane B. Eisenman, Emiliano Miluzzo, Nicholas D. Lane, Ronald A. Peterson, Gahng-Seop Ahn, and Andrew T. Campbell. Bikenet: A mobile sensing system for cyclist experience mapping. ACM Trans. Sen. Netw., 6(1):6:1–6:39, January 2010.

[71] V. Erceg, L.J. Greenstein, S. Tjandra, S.R. Parkoff, A. Gupta, B. Kulic, A. Julius, and R. Jastrzab. An empirically-based path loss model for wireless channels in suburban environments. In GLOBECOM'98, volume 2, pages 922–927, 1998.

[72] V. Erceg, K.V.S. Hari, and et al. Channel models for fixed wireless applications. Technical report, IEEE 802.16 Broadband Wireless Access Working Group, 2001.

[73] G. Evans, B. Joslin, L. Vinson, and B. Foose. The optimization and application of the w.c.y. lee propagation model in the 1900 mhz frequency band. In Vehicular Technology Conference, 1997 IEEE 47th, volume 1, pages 87 –91 vol.1, May 1997.

[74] National Science Foundation. National center for airborne laser mapping. http://www.ncalm.cive.uh.edu/, March 2012.

[75] The Open Source Geospatial Foundation. Geospatial Data Abstraction Library (GDAL). http://www.gdal.org/.

[76] C. Fretzagias and M. Papadopouli. Cooperative location-sensing for wireless networks. In Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on, pages 121 – 131, 2004.

[77] Harald T. Friis. A note on a simple transmission formula. In Proceedings of the IRE and Waves and Electrons, May 1946.

[78] Vijay K. Garg. Wireless Communications and Networking. Morgan Kaufmann, 2007.

[79] Sean Gillies, Aron Bierbaum, Kai Lautaportti, and Oliver Tonnhofer. Shapely 1.2.14. http://pypi.python.org/pypi/Shapely, March 2012.

[80] N.C. Goncalves and L.M. Correia. A propagation model for urban microcellular systems at the uhf band. Vehicular Technology, IEEE Transactions on, 49(4):1294 –1302, July 2000.

[81] V. Goncalves and S. Pollin. The value of sensing for tv white spaces. In New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2011 IEEE Symposium on, pages 231 –241, may 2011.

[82] Inc. Google. Google earth. http://www.google.com/earth/, April 2012.

[83] Google Inc. Google wifi for mountain view. http://wifi.google.com/, March 2011.

[84] D.B. Green and A.S. Obaidat. An accurate line of sight propagation performance model for ad-hoc 802.11 wireless lan (wlan) devices. In Communications, 2002. ICC 2002. IEEE International Conference on, volume 5, pages 3424 – 3428 vol.5, 2002.

[85] L.J. Greenstein and V. Erceg. Gain reductions due to scatter on wireless paths with directional antennas. IEEE Communications Letters, 3(6):169–171, 1999.

[86] L.J. Greenstein, V. Erceg, Y.S. Yeh, and M.V. Clark. A new path-gain/delay-spread propagation model for digital cellular channels. Vehicular Technology, IEEE Transactions on, 46(2):477 –485, may 1997.

[87] Shane Grigsby. Open source lidar visualization using grass gis. In Free Open Source Software for Geographical Information Systems (FOSS4G), 2011.

[88] PostgreSQL Global Development Group. Postgresql: The world's most advanced open source database. http://www.postgresql.org/, June 2011.

[89] D. Gurney, G. Buchwald, L. Ecklund, S.L. Kuffner, and J. Grosspietsch. Geo-location database techniques for incumbent protection in the tv white space. In New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on, pages 1 –9, oct. 2008.

[90] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Predictable 802.11 packet delivery from wireless channel measurements. In ACM SIGCOMM, 2010.

[91] Kate Harrison, Shridhar Mubaraq Mishra, and Anant Sahai. How much white-space capacity is there? In DySPAN'10, 2010.

[92] Shousheng He and M. Torkelson. Effective snr estimation in ofdm system simulation. In Global Telecommunications Conference, 1998. GLOBECOM 98. The Bridge to Global Integration. IEEE, volume 2, pages 945 –950 vol.2, 1998.

[93] Kristina Helle and Edzer Pebesma. Optimizing spatio-temporal sampling designs of synchronous, static, or clustered measurements. Geophysical Research Abstracts, 12, 2010.

[94] K.T. Herring, J.W. Holloway, D.H. Staelin, and D.W. Bliss. Path-loss characteristics of urban wireless channels. IEEE Transactions on Antennas and Propagation, 58(1):171 –177, jan. 2010.

[95] A. Hills. Large-scale wireless lan design. Communications Magazine, IEEE, 39(11):98–107, Nov 2001.

[96] A. Hills and J. Schlegel. Rollabout: a wireless design tool. Communications Magazine, IEEE, 42(2):132 – 138, February 2004.

[97] Jeremy Hsu. Wi-fi cloud hovers over salt lake city. IEEE Spectrum Online, February 2008. http://spectrum.ieee.org/feb08/6025.

[98] G.A. Hufford. The ITS irregular terrain model, version 1.2.2, the algorithm. `http://flattop.its.bldrdoc.gov/itm.html`.

[99] G.A. Hufford, A.G. Longley, and W.A. Kissick. A guide to the use of the ITS irregular terrain model in the area prediction mode. Technical Report 82-100, NTIA, 1982.

[100] F. Ikegami, T. Takeuchi, and S. Yoshida. Theoretical prediction of mean field strength for urban mobile radio. IEEE Transactions on Antennas and Propagation, 39:299–302, 1991.

[101] Staircase 3 Inc. Open signal maps. http://opensignalmaps.com/, April 2012.

[102] The Khronos Group. Sony Computer Entertainment Inc. Collada digital asset and fx exchange schema. http://collada.org/, January 2012.

[103] Intermap Technologies. Digital surface models. http://www.intermap.com/digital-surface-models, January 2011.

[104] Edward H Isaaks and R. Mohan Srivastava. An Introduction to Applied Geostatistics. Oxford University Press, 1989.

[105] Magdy F. Iskander and Zhengqing Yun. Propagation prediction models for wireless communication systems. IEEE Transactions on Vehicular Technology, 50:662–672, 2002.

[106] ITU-R. Digital topographic databases for propagation studies. Technical Report P.1058, ITU, 1999.

[107] ITU-R. Terrestrial land mobile radiowave propagation in the VHF/UHF bands. ITU-R, 2002.

[108] ITU-R. Attenuation in vegetation. Technical Report P.833, ITU, 2005.

[109] ITU-R. Prediction procedure for the evaluation of microwave interference between stations on the surface of the earth at frequencies above about 0.7 ghz. Technical Report P.452, ITU, 2007.

[110] ITU-R. Attenuation by atmospheric gases. Technical Report 676-8, ITU, 2009.

[111] JDSU. Base station analyzer: Lte phy layer measurement guide. Application note, JDSU, 2011.

[112] Xiaodong Jian, Ricardo A. Olea, and Yun-Sheng Yu. Semivariogram modeling by weighted least squares. Computers & Geosciences, 22(4):387 – 397, 1996.

[113] A. Journel and R. Froidevaux. Anisotropic hole-effect modeling. Mathematical Geology, 14:217–239, 1982. 10.1007/BF01032885.

[114] Paulo J. Ribeiro Jr and Peter J. Diggle. geoR: a package for geostatistical analysis. R-NEWS, 1(2):14–18, June 2001. ISSN 1609-3631.

[115] M. Kamenetsky and M. Unbehaun. Coverage planning for outdoor wireless lan systems. Broadband Communications, 2002. Access, Transmission, Networking. 2002 International Zurich Seminar on, pages 49–1–49–6, 2002.

[116] Varun Kanade and Santosh Vempala. Life (and routing) on the wireless manifold. In ACM SigComm HotNets, 2007.

[117] Athanasios G. Kanatas, Ioannis D. Kountouris, George B. Kostaras, and Philip Constantinou. A utd propagation model in urban microcellular environments. IEEE Transactions on Vehicular Technology, 46:185–193, 1997.

[118] Mikhail Kanevski, editor. Advanced Mapping of Environmental Data. ISTE Ltd. and John Wiley and Sons, Inc., 2008.

[119] Abdullah Konak. A kriging approach to predicting coverage in wireless networks. International Journal of Mobile Network Design and Innovation, 2010.

[120] David Kotz, Calvin Newport, and Chip Elliot. The mistaken axioms of wireless-network research. Technical report, Dartmouth College of Computer Science, 2003.

[121] Lance Kramer. Logan kleier: Frustrated with the city's free wireless? portland's wi-fi man explains why you should be patient. Willamette Week, March 28th 2007.

[122] R. D. J. Kramer, A. Lopez, and A. M. J. Koonen. Municipal broadband access networks in the netherlands - three successful cases, and how new europe may benefit. In AcessNets '06: Proceedings of the 1st international conference on Access networks, page 12, New York, NY, USA, 2006. ACM.

[123] D. G. Krige. A statistical approach to some basic mine valuation problems on the witwatersrand. Journal of the Chemical, Metallurgical, and Mining Society of South Africa, 1952.

[124] Danie Krige and Wynand Kleingeld. The genesis of geostatistics in gold and diamond industries. In P. Bickel, P. Diggle, S. Fienberg, U. Gather, I. Olkin, S. Zeger, Michel Bilodeau, Fernand Meyer, and Michel Schmitt, editors, Space, Structure and Randomness, volume 183 of Lecture Notes in Statistics, pages 5–16. Springer New York, 2005.

[125] L-COM. Antenna specifications. http://www.l-com.com/, January 2011.

[126] N.D. Lane, E. Miluzzo, Hong Lu, D. Peebles, T. Choudhury, and A.T. Campbell. A survey of mobile phone sensing. Communications Magazine, IEEE, 48(9):140 –150, sept. 2010.

[127] R. M. Lark. Estimating variograms of soil properties by the method-of-moments and maximum likelihood. European Journal of Soil Science, 51:717–728, 2000.

[128] R. M. Lark. Optimized spatial sampling of soil for estimation of the variogram by maximum likelihood. Geoderma, 105(1-2):49 – 80, 2002.

[129] Ian W.C. Lee and Abraham O. Fapojuwo. Stochastic processes for computer network traffic modeling. Computer Communications, 29(1):1–23, December 2005.

[130] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. Slaw: A new mobility model for human walks. In INFOCOM 2009, IEEE, pages 855 –863, april 2009.

[131] W.C.Y. Lee. Lee's model [cellular radio path loss prediction]. In Vehicular Technology Conference, 1992, IEEE 42nd, pages 343 –348 vol.1, May 1992.

[132] William C. Y. Lee. Estimate of local average power of a mobile radio signal. IEEE Transactions on Vehicular Technology, VT-34:22–27, 1985.

[133] William C. Y. Lee and David J. Y. Lee. Microcell prediction in dense urban area. IEEE Transactions on Vehicular Technology, 47:246–253, 1998.

[134] Le-Wei Li, Tat-Soon Yeo, Pang-Shyan Kooi, and Mook-Seng Leong. Radio wave propagation along mixed paths through a four-layered model of rain forest: an analytic approach. Antennas and Propagation, IEEE Transactions on, 46(7):1098 –1111, jul 1998.

[135] Lorne C. Liechty, Eric Reifsnider, and Greg Durgin. Developing the best 2.4 ghz propagation model from active network measurements. In VTC'07, 2007.

[136] John A. Magliacane. SPLAT! A Terrestrial RF Path Analysis Application for Linux/Unix, March 2008.

[137] Nicolas Maisonneuve, Matthias Stevens, Maria E. Niessen, and Luc Steels. Noisetube: Measuring and mapping noise pollution with mobile phones. In R. Allan, U. Frstner, and W. Salomons, editors, Information Technologies in Environmental Engineering, Environmental Science and Engineering, pages 215–228. Springer Berlin Heidelberg, 2009.

[138] Maptools.org. Shapefile c library v1.2. http://shapelib.maptools.org/, March 2012.

[139] B.P. Marchant and R.M. Lark. Optimized sample schemes for geostatistical surveys. Mathematical Geology, 39:113–134, 2007.

[140] Sue Marek. In photos: Riding shotgun with the verizon network tester. http://www.fiercewireless.com/story/photos-riding-shotgun-verizon-network-tester/2011-06-17, June 2011.

[141] Chairman Martin, the Commissioners Copps, Adelstein, McDowell, and Tate. In the matter of unlicensed operation in the TV broadcast bands. additional spectrum for unlicensed devices below 900 MHz and in the 3 GHz band. Technical Report FCC 08-260, FCC, 2008.

[142] Paul McKenna. National technology industry association: Institute for telecommunication sciences. Personal Communication, January 2011.

[143] A. Medeisis and A. Kajackas. On the use of the universal Okumura-Hata propagation prediction model in rural areas. In IEEE VTC'00, 2000.

[144] Nicholas Metropolis, Ariana W. Reosenbluth, Marshall N. Rosenbluth, and Augusta H. Teller. Equation of state calculations by fast computing machines. The Journal of Chemical Physics, 21(6):1087–1092, June 1953.

[145] III Mitola, J. and Jr. Maguire, G.Q. Cognitive radio: making software radios more personal. Personal Communications, IEEE, 6(4):13 –18, aug 1999.

[146] MuniWireless. City and County-wide WiFi Networks. `http://www.muniwireless.com/city-county-wifi-networks-page/`, December 2010.

[147] Alan T. Murray and Richard L. Church. Applying simulated annealing to location-planning models. Journal of Heuristics, 2:31–53, 1996.

[148] M. Nakagami. Statistical Methods in Radio Wave Propagation, chapter The m-distribution - A general formula of intensity distribution of rapid fading, pages 3–36. Program Press, Oxford, 1960.

[149] Vishnu Navda, Anand Prabhu Subramanian, Kannan Dhanasekaran, Andreas Timm-Giel, and Samir R. Das. Mobisteer: using steerable beam directional antenna for vehicular network access. In Edward W. Knightly, Gaetano Borriello, and Ramón Cáceres, editors, MobiSys, pages 192–205. ACM, 2007.

[150] A. Neskovic, N. Neskovic, and D. Paunovic. Indoor electric field level prediction model based on the artificial neural networks. Communications Letters, IEEE, 4(6):190 –192, June 2000.

[151] Aleksandar Neskovic, Natasa Neskovic, and George Paunovic. Modern approaches in modeling of mobile radio systems propagation environment. IEEE Communications Surveys and Tutorials, 3(3), 2000.

[152] D.E. Newbury and D.S. Bright. Logarithmic 3-band color encoding: Robust method for display and comparison of compositional maps in electron probe x-ray microanalysis. Technical report, Surface and Microanalysis Science Division, National Institute of Standards and Technology (NIST), 1999.

[153] NIST. `http://w3.antd.nist.gov/wctg/manet/calcmodels_r1.pdf`.

[154] Yasuhiro Oda and Koichi Tsunekawa. Advances los path loss model in microwave mobile communications. In 10th International Conference on Antennas and Propagation, 1997.

[155] University of Colorado. Environmental center. http://ecenter.colorado.edu/, March 2012.

[156] University of Colorado Boulder: Research Computing. Janus supercomputer. https://www.rc.colorado.edu/systems/supercomputer, December 2011. This work utilized the Janus supercomputer, which is supported by the National Science Foundation (award number CNS-0821794) and the University of Colorado Boulder. The Janus supercomputer is a joint effort of the University of Colorado Boulder, the University of Colorado Denver and the National Center for Atmospheric Research.

[157] Y. Okumura, E. Ohmori, T. Kawano, , and K. Fukuda. Field strengh and its variability in VHF and UHF land mobile radio service. Rev. Electr. Commun. Lab., 16:825–73, 1968.

[158] Ricardo Olea. Sampling design optimization for spatial functions. Mathematical Geology, 16:369–392, 1984.

[159] Ricardo Olea. A six-step practical approach to semivariogram modeling. Stochastic Environmental Research and Risk Assessment, 20:307–318, 2006.

[160] OSGeo Project. PostGIS. http://postgis.refractions.net/, June 2011.

[161] Y. Ould Isselmou, H. Wackernagel, W. Tabbara, and J. Wiart. Geostatistical interpolation for mapping radio-electric exposure levels. In Antennas and Propagation, 2006. EuCAP 2006. First European Conference on, pages 1 –6, 2006.

[162] Eulogio Pardo-Igízquiza. Mlreml: A computer program for the inference of spatial covariance parameters by maximum likelihood and restricted maximum likelihood. Computers & Geosciences, 23:153–162, 1997.

[163] Eulogio Pardo-Igízquiza. Maximum likelihood estimation of spatial covariance parameters. Mathematical Geology, 30(1):95–108, 1998.

[164] Corey Pein. The hole in the fiber doughnut: Commissioner dan saltzman wants fiber in the citys web diet. Willamette Week, July 23rd 2008.

[165] Hagen Paul Pfeifer. On the validation of radio propagation models. `http://jauu.net/data/pdf/propagation-models.pdf`, January 2010.

[166] Caleb Phillips and Eric W. Anderson. CRAWDAD data set cu/cuwart (v. 2011-10-24). Downloaded from http://crawdad.cs.dartmouth.edu/cu/cuwart, October 2011.

[167] Caleb Phillips, Scott Raynel, Jamie Curtis, Sam Bartels, Douglas Sicker, Dirk Grunwald, and Tony McGregor. The efficacy of path loss models for fixed rural wireless links. In Passive and Active Measurement Conference (PAM), 2011.

[168] Caleb Phillips and Russell Senior. Unwire portland proof-of-concept network testing. Technical report, unwirepdx-watch.org, 2007.

[169] Caleb Phillips and Russell Senior. Unwire-PDX Watch, March 2008.

[170] Caleb Phillips and Russell Senior. CRAWDAD data set pdx/metrofi (v. 2011-10-24). Downloaded from http://crawdad.cs.dartmouth.edu/pdx/metrofi, October 2011.

[171] Caleb Phillips, Russell Senior, Douglas Sicker, and Dirk Grunwald. Robust coverage and performance testing for large area networks. In AccessNets, 2008.

[172] Caleb Phillips, Douglas Sicker, and Dirk Grunwald. Bounding the error of path loss models. In IEEE Dynamic Spectrum Access Networks (DySPAN), May 2011.

[173] Caleb Phillips, Douglas Sicker, and Dirk Grunwald. The stability of the longley-rice irregular terrain model for typical problems. Technical Report CU-CS-1086-11, University of Colorado at Boulder Computer Science Department, 2011.

[174] Caleb Phillips, Douglas Sicker, and Dirk Grunwald. Bounding the practical error of path loss models. International Journal of Antennas and Propagation, 2012, 2012.

[175] Caleb Phillips, Douglas Sicker, and Dirk Grunwald. A survey of wireless path loss prediction and coverage mapping methods. IEEE Communications Society Surveys and Tutorials, 2012.

[176] Leonard Piazzi and Henry L. Bertoni. Achievable accuracy of site-specific path-loss predictions in residential environments. IEEE Transactions on Vehicular Technology, 48:922–930, 1999.

[177] OSGeo Project. Quantum gis. http://www.qgis.org/, March 2012.

[178] M.J. Pyrcz and C.V. Deutsch. The whole story on the hole effects. Technical report, Geostatistical Association of Australasia, 2003.

[179] GNU Radio. Gnu radio. http://gnuradio.org/redmine/projects/gnuradio/wiki, December 2011.

[180] Bhaskaran Raman and Kameswari Chebrolu. Experiences in using wifi for rural internet in india. Communications Magazine, IEEE, 45(1):104 –110, 2007.

[181] Ram Ramanathan. On the performance of ad hoc networks with beamforming antennas. In Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking and computing, pages 95–105, Long Beach, CA, USA, 2001. ACM Press.

[182] Sampath Rangarajan. Geni - open, programmable wimax base station. http://www.winlab.rutgers.edu/docs/focus/GENI-WiMAX.html, August 2011.

[183] Theodore S. Rappaport. Wireless Communications: Principles and Practice. Prentice Hall, 2002.

[184] Shravan Rayanchu, Ashish Patro, and Suman Banerjee. Airshark: Detecting non-wifi rf devices using commodity wifi hardware. In Internet Measurement Conference (IMC), November 2–4 2011.

[185] Scott Raynel. WMP - The Wireless Measurement Project. http://www.wand.net.nz/ smr26/wmp/, November 2010.

[186] Brad Reed. Lte vs. wimax. http://www.networkworld.com/news/2011/102011-tech-arguments-lte-wimax-252225.html, November 2011.

[187] REMCOM. Electromagnetic simulation solutions. http://www.remcom.com/, January 2012.

[188] REMCOM. Engineering. Personal Communication, January 2012.

[189] REMCOM. Wireless insight: Wireless em propagation software. http://www.remcom.com/wireless-insite, January 2012.

[190] M. Riback, J. Medbo, J.-E. Berg, F. Harrysson, and H. Asplund. Carrier frequency effects on path loss. In VTC 2006-Spring, volume 6, pages 2717 –2721, 7-10 2006.

[191] Rice University. Google wi fi. http://tfa.rice.edu/measurements /index.php?n=Main.GoogleWiFi, March 2011.

[192] Rice University. Tfa measurements. http://tfa.rice.edu/measurements/, March 2011.

[193] J. Riihijarvi, P. Mahonen, M. Petrova, and V. Kolar. Enhancing cognitive radios with spatial statistics: From radio environment maps to topology engine. In CROWNCOM '09, pages 1 –6, June 2009.

[194] J. Riihijarvi, P. Mahonen, and S. Sajjad. Influence of transmitter configurations on spatial statistics of radio environment maps. In Personal, Indoor and Mobile Radio Communications, pages 853 –857, Sep 2009.

[195] J. Riihijarvi, P. Mahonen, M. Wellens, and M. Gordziel. Characterization and modelling of spectrum for dynamic spectrum access with spatial statistics and random fields. In Personal, Indoor and Mobile Radio Communications (PIMRC 2008), pages 1–6, Sep 2008.

[196] Brian D. Ripley. Statistical Inference for Spatial Processes. Cambridge University Press, 1988.

[197] Brian D. Ripley. Spatial Statistics. Wiley Interscience, 2004.

[198] K. Rizk, J.-F. Wagen, and F. Gardiol. Two-dimensional ray-tracing modeling for propagation prediction in microcellular environments. IEEE Transactions on Vehicular Technology, 46(2):508 –518, May 1997.

[199] Joshua Robinson. Personal communication. Email, March 2011.

[200] Joshua Robinson, Ram Swaminathan, and Edward W. Knightly. Assessment of urban-scale wireless networks with a small number of measurements. In MobiCom, 2008.

[201] Bernice E. Rogowitz and Lloyd A. Treinish. Why should engineers and scientists be worried about color? Technical report, IBM Thomas J. Watson Research Center, 1996.

[202] A. J. Rustako, Noach Amitay, G. J. Owens, and R. S. Roman. Radio propagation at microwave frequencies for line-of-sight microcellular mobile and personal communications. IEEE Transactions on Vehicular Technology, 40:203–210, 1991.

[203] S. R. Saunders and F. R. Bonar. Explicit multiple building diffraction attenuation function for mobile radio wave propagation. Electronics Letters, 27(14):1276–1277, July 1991.

[204] Nmap Security Scanner. Nping. http://nmap.org/nping/, March 2012.

[205] S.Y. Seidel and T.S. Rappaport. Path loss prediction in multifloored buildings at 914 mhz. Electronics Letters, 27(15):1384 –1387, 1991.

[206] John S. Seybold. Introduction to RF Propagation. Wiley Interscience, 2005.

[207] Purnima K. Sharma and R.K. Singh. Comparative analysis of propagation path loss models with field measured databases. International Journal of Engineering Science and Technology, 2:2008–2013, 2010.

[208] Hweechul Shin. Measurements and models of 802.11b signal strength variation over small distances. Master's thesis, University of Delaware, 2010.

[209] Google Sketchup. 3d warehouse. http://sketchup.google.com/3dwarehouse/, January 2012.

[210] B. Sklar. Rayleigh fading channels in mobile digital communication systems .i. characterization. Communications Magazine, IEEE, 35(7):90 –100, July 1997.

[211] Vinay Sridhara and Stephan Bohacek. Realistic propagation simulation of urban mesh networks. Computer Networks, 51(12):3392 – 3412, 2007.

[212] Staircase3. Staircase3. http://staircase3.com/, April 2012.

[213] H. Steendam and M. Moeneclaey. Analysis and optimization of the performance of ofdm on frequency-selective time-selective fading channels. Communications, IEEE Transactions on, 47(12):1811 –1819, dec 1999.

[214] Stephen V. Stehman. Basic probability sampling designs for thematic map accuracy assessment. International Journal of Remote Sensing, 20(12):2423–2441, 1999.

[215] Anand Prabhu Subramanian, Pralhad Deshpande, Jie Gao, and Samir R. Das. Drive-by localization of roadside WiFi networks. In 27th Annual IEEE Conference on Computer Communications (INFOCOM 2008), Phoenix, Arizona, April 2008.

[216] Tammy Sun. Spokesperson statement on ntia letter–lightsquared and gps. http://www.fcc.gov/document/spokesperson-statement-ntia-letter-lightsquared-and-gps, February 2012.

[217] Diane Tang and Mary Baker. Analysis of a metropolitan-area wireless network. Wireless Networks, 8(2-3):107–120, 2002.

[218] Hongsuda Tangmunarunkit, Ramesh Govindan, Sugih Jamin, Scott Shenker, and Walter Willinger. Network topology generators: degree-based vs. structural. In SIGCOMM '02, pages 147–159, New York, NY, USA, 2002. ACM.

[219] TCPDump. Tcpdump & libpcap. http://www.tcpdump.org/, April 2012.

[220] The Open MPI Team. Open mpi: Open source high performance computing. http://www.open-mpi.org/, December 2011.

[221] Riverbed Technology. Windump: tcpdump for windows using winpcap. http://www.winpcap.org/windump/, April 2012.

[222] Technology for All. TFA-Wireless. http://www.techforall.org/Programs/ ResearchandInnovation/TFAWireless/tabid/107/Default.aspx, March 2011.

[223] TIA. Wireless communications systems - performance in noise and interference - limited situations - recommended methods for technology - independent modeling, simulation, and verifications. Technical Report TSB-88-B-1, May 2005.

[224] Saúl Torrico, Henry L. Bertoni, and Roger H. Lang. Modeling tree effects on path loss in a residential environment. IEEE Transactions on Antennas and Propagation, 46(6):872–880, 1998.

[225] Trilliant Inc. SkyPilot. http://skypilot.trilliantinc.com/, January 2011.

[226] Tropos Networks Inc. Tropos networks. http://www.tropos.com/, March 2011.

[227] United States Federal Communications Commission. Understanding wireless telephone coverage areas: Guide. http://www.fcc.gov/guides/understanding-wireless-telephone-coverage-areas, July 2011.

[228] Ian Urbina. Hopes for wireless cities fade as internet providers pull out. The New York Times, (22), March 2008.

[229] V. Valenta, R. Mars anda andlek, G. Baudoin, M. Villegas, M. Suarez, and F. Robert. Survey on spectrum utilization in europe: Measurements, analyses and observations. In Cognitive Radio Oriented Wireless Networks Communications (CROWNCOM), 2010 Proceedings of the Fifth International Conference on, pages 1 –5, june 2010.

[230] J.W. van Groenigen, W. Siderius, and A. Stein. Constrained optimisation of soil sampling for minimisation of the kriging variance. Geoderma, 87(3-4):239 – 259, 1999.

[231] Lewis E. Vogler. An attenuation function for multiple knife-edge diffraction. Radio Science, 17(6):1541–1546, 1982.

[232] Esme Vos. Metrofis portland network to shut down. http://www.muniwireless.com/2008/06/23/metrofis-portland-network-to-shut-down/, June 2008.

[233] Hans Wackernagel. Multivariate Geostatistics. Springer, 2nd edition.

[234] René Wahl, Gerd Wölfe, Philipp Wertz, Pascal Wildbolz, and Friedlich Landstorfer. Dominant path prediction model for urban scenarios. In 14th IST Mobile and Wireless Communications Summit, 2005.

[235] J. Walfisch and H.L. Bertoni. A theoretical model of uhf propagation in urban environments. IEEE Transactions on Antennas and Propagation, 36(12):1788 –1796, dec. 1988.

[236] Matthias Wellens, Janne Riihijärvi, and Petri Mähönen. Spatial statistics and models of spectrum use. Comput. Commun., 32:1998–2011, December 2009.

[237] J.H. Whitteker. Physical optics and field-strength predictions for wireless systems. Selected Areas in Communications, IEEE Journal on, 20(3):515 –522, apr 2002.

[238] EDX Wireless. Engineering. Personal Communication, January 2012.

[239] EDX Wireless. Smart planning for smart networks. http://www.edx.com/, January 2012.

[240] Verizon Wireless. Personal communication. Email, September 2011.

[241] G. Wölfle, R. Hoppe, and F.M. Landstorfer. Radio network planning with ray optical propagation models for urban, indoor, and hybrid scenarios. In 11th IEEE Wireless Conference, 1999.

[242] H.H. Xia and H.L. Bertoni. Diffraction of cylindrical and plane waves by an array of absorbing half-screens. Antennas and Propagation, IEEE Transactions on, 40(2):170 –177, February 1992.

[243] Zhe Xiang, Hangjin Zhang, Jian Huang, Song Song, and K.C. Almeroth. A hidden environment model for constructing indoor radio maps. In World of Wireless Mobile and Multimedia Networks (WoWMoM 2005), pages 395 – 400, Jun 2005.

[244] Emre A. Yavuz and Gunnar Karlsson. Poster: collective mobile sensing for wireless network performance monitoring. In MobiSys '11 Proceedings of the 9th international conference on Mobile systems, applications, and services, 2011.

[245] Evangelos A. Yfantis, George T. Flatman, and Joseph V. Behar. Efficiency of kriging estimation for square, triangular, and hexagonal grids. Mathematical Geology, 19(3):183–205, 1987.

[246] N. Youssef, T. Munakata, and M. Takeda. Fade statistics in nakagami fading environments. In IEEE 4th International Symposium on Spread Spectrum Techniques and Applications Proceedings, volume 3, pages 1244–1247, 1996.

[247] Ellen W. Zegura, Ken Calvert, and S. Bhattacharjee. How to model an internetwork. In Infocom. IEEE, 1996.

[248] Wei Zhang and N. Moayeri. Closed-form expressions for the prediction of microcellular mobile radio propagation in urban environments. IEEE Transactions on Antennas and Propagation, 51(5):952 – 956, may 2003.

[249] Stanislav Zvanovec, Pavel Pechac, and Martin Klelpal. Wireless lan networks design: Site survey or propagation modeling? Radioengineering, 12(4):42–49, December 2003.

Appendix A

EDAM: The Effective Directivity Antenna Model

Increasingly, wireless networks are using directional antennas to improve the throughput, reach of networks [181], or to reduce interference between adjacent networks and other noise sources. A more recent development is the use of electronically steerable directional or phase array antennas [149, 33, 215]. These antennas provide better network performance by dynamically controlling the radiation pattern of the antenna. Networks that utilize these antennas can reap substantial improvements in efficiency at all layers of the networking stack.

Different network simulators model such antennas with different degrees of fidelity. This chapter argues that the models in the most common network simulators make such simplifying assumptions that it is often difficult to draw strong conclusions from the simulations derived using those models. This is demonstrated this using a series of measurements with several different and widely used directional antenna configurations. A more accurate model is developed based on measurements and intuitions about radio propagation[1]. This model captures more about the uncertainty of the environment than the specifics of the antenna and that our results should be generally applicable to many different directional antenna patterns with similar gain characteristics.

The measurement study described here uses sophisticated measurement equipment, including a vector signal analyzer (VSA) and signal generator (VSG). Since the costs of such equipment are prohibitive, a method that uses inexpensive equipment (such as standard networking cards) is also developed to produce

[0] Work in this appendix has also been published in [26, 25, 27, 28, 30].

[1] All of the measurements collected for this research are available publicly at [6]. An implementation of our model for the Qualnet 4.5.1 simulator is available at `http://systems.cs.colorado.edu`.

the data needed for the derived models.

A.1 Directional Models

The simulators commonly used in networking research do not consider antenna directionality and radio propagation as interacting variables. This paper considers three widely used simulators, *OpNet*, *QualNet*, and *NS-2*. Each one supports several models of radio propagation, but they all follow the same general model with regard to antenna gain: For any two stations i and j, the received signal strength is computed according to the general form of equation A.1:

$$\text{Received Power} = P_{tx} * G_{tx} * |PL(i,j)| * G_{rx} \tag{A.1}$$

The received power P_{rx} is the product of the transmitted power P_{tx}, the transmitter's gain G_{tx}, the magnitude of path loss between the two stations $|PL(i,j)|$, and the receiver's gain G_{rx}.

The transmitter and receiver gains are treated as constants in the case of omnidirectional (effectively isotropic in the azimuth plane) antennas. For directional antennas, however, gain is an antenna-specific function of the direction of interest. The orientation of an antenna can be modeled in terms of its zenith (ϕ) and azimuth (θ). Then, for a given antenna a, characterization function $f_a(\phi, \theta)$ can be defined:

$$\text{Gain in direction } (\phi, \theta) = f_a(\phi, \theta) \tag{A.2}$$

$$\text{Combined gain} = f_a(\phi, \theta) * f_b(\phi', \theta') \tag{A.3}$$

Correspondingly, the receiver gain is modeled by a (potentially different) function of the direction from which the signal is received. Besides being a source of interference for a dominant signal, the energy traveling along secondary paths also carries signal. If one of the weaker signals for a transmitter happens to be aligned with a high gain direction of a receiving antenna, the received power from that path can be greater than that of the primary path.

The above models describe the power emitted in, or received from, a single direction. In reality, the transmitter's power is radiated in all directions, and the receiver aggregates power (be it signal or noise) from all directions. Although the simulators considered here assume that the single direction of interest for each

station is precisely toward the other station, equations A.1 and A.3 can be generalized to the case where there are multiple significant signal paths:

$$P_{rx} = \sum_{l \in \text{paths}} P_{tx} * f_a(\phi_l, \theta_l) * PL_l(i,j) * f_b(\phi'_l, \theta'_l) \tag{A.4}$$

In Equation A.4, note that that P_{rx} is not necessarily all "signal". It may be the case that only one signal is decodable and the others destructively interfere. In this case equation A.5 is a better model:

$$P_{rx} = \max_{l \in \text{paths}} P_{tx} * f_a(\phi_l, \theta_l) * PL_l(i,j) * f_b(\phi'_l, \theta'_l) \tag{A.5}$$

Both of these models assume that there is some way to describe available paths that a signal may take. As with the Rayleigh and Rician fading models, it may be possible to build a parameterized model of those paths for "cluttered" and "uncluttered" environments. This is the approach taken here, using measured data to determine the model.

With any of the three simulators we consider, the user has the freedom to provide any type of mapping between gain and angle. This means that the user could conceivably make measurements with their desired hardware in their desired environment, much as we have done, and then install this as the pattern. However, even though the antenna can conceivably be modeled arbitrarily well, the *directionality of the signal* is an effect of the interaction between antenna and environment and that modeling both in isolation, however well, misses significant effects. This chapter proposes a combined empirical model that attempts to account for both the pattern of the antenna and the deviation from this pattern due to environmental effects.

A.2 Method

This section describes the method devised for deriving empirical models for antenna patterns using commodity hardware and address any reservations about their accuracy by providing a means for equipment calibration.

A.2.1 Data Collection Procedure

Two laptops are used, one configured as a receiver and the other as a transmitter. Each is equipped with an Atheros-based MiniPCI-Express radio that is connected to an external antenna using a U.Fl to N pigtail adapter and a length of LMR-400 low loss antenna cable. The receiver laptop is connected to a 7 dBi omnidirectional antenna on a tripod approximately two meters off the ground. The transmitter laptop is connected to the antenna we intend to model on a tripod 30.5 m from the receiver, also two meters off the ground. The transmitter tripod features a geared triaxial head, which allows precise rotation.

The transmitter radio is put in 802.11x ad hoc mode on the least congested channel. The transmitter's ARP table is manually hacked to allow it to send UDP packets to a nonexistent receiver. The receiver is put in monitor mode on the same channel and logs packets with tcpdump. Finally, both the receiver and transmitter must have antenna diversity disabled. With the equipment in place, the procedure is as follows: For each 5 degree position about the azimuth, send 500 unacknowledged UDP packets. Without intervention otherwise, due to MAC-layer retransmits, each will be retried 8 times, resulting in 4000 distinct measurements.

During the experiment, the researchers themselves must be careful to stay well out of the nearfield of the antennas and to move to the same location during runs (so that they, in effect, become a static part of the environment). If additional data is desired for a given location, multiple receivers can be used, provided the data from them is treated separately (as each unique path describes a unique environment).

In the process of collection, some packets will be dropped due to interference or poor signal. In practice, the percentage of dropped frames *per angle* is very small: the maximum lost frames per angle in the data sets is on the order of 5%, with less than 1% lost being more common (the mean is 0.01675%). Moreover, the correlation coefficient between angle and loss percentage is -0.0451, suggesting that losses are uniformly distributed across angles. Given that 4000 samples have been taken in each direction, noise in the measurements due to packet loss is negligible.

A.3 Measurements

This section explains the data sets collected, discusses the normalization procedure developed, and gives some high level statistical characterization of the data.

A.3.1 Experiments Performed

In order to derive an empirical model that better fits real world behavior, data was collected in several disparate environments with three different antennas. A summary of these data sets is provided in table A.1. With the exception of the reference patterns, all of the measurements were made with commodity hardware by sending many measurement packets between two antennas and logging received signal strength (RSS) at the receiver. The three antenna configurations used include: (1) a HyperLink 24dBi parabolic dish with an 8 degree horizontal beamwidth, (2) a HyperLink 14dBi patch with a 30 degree horizontal beamwidth, and (3) a Fidelity Comtech Phocus 3000 8-element uniform circular phased array with a main lobe beamwidth of approximately 52 degrees. This phased array functions as a switched beam antenna and can form this beam in one of 16 directions (on 22.5 degree increments around the azimuth). For the HyperLink antennas, the same antenna was used in all experiments of a particular type to avoid intra-antenna variation due to manufacturing differences.

In addition to the *in situ* experiments, a "reference" data set is available for each configuration. The Array-Reference data set was provided to us by the antenna manufacturer. Because HyperLink could not provide us with data on their antennas, Parabolic-Reference and Patch-Reference were derived using an Agilent 89600S VSA and an Agilent E4438C VSG in a remote floodplain[2].

Following is a brief description of each of the experiments:

Parabolic-Outdoor-A, Patch-Outdoor-A: A large open field on the University of Colorado campus was used for these experiments. The field is roughly 150m on a side and is surrounded by brick buildings on two of the four sides. Although there is line-of-sight and little obstruction, the surrounding structures make this

[2] We were unable to aquire access to an anechoic chamber in time for this study, but would like to make use of one in future work, for even cleaner reference measurements.

location most representative of an urban outdoor deployment.

Parabolic-Outdoor-B, Patch-Outdoor-B: A large University-owned floodplain on the edge of town was used for the most isolated data sets. The floodplain is flat, recessed, and is free from obstruction for nearly a quarter mile in all directions. This location is most representative of a rural backhaul link.

Array-Outdoor-A: The same open field is used as in the Parabolic-Outdoor-A and Patch-Outdoor-A data sets. The collection method here differs from that described in section A.2. A single phased array antenna is placed approximately 30 m away from an omnidirectional transmitter. The transmitter sends a volley of packets from its fixed position as the phased array antenna electronically steers its antenna across each of its 16 states, spending 20 ms in each state. Several packets are received in each directional state. The phased array antenna is then manually rotated in 10 degree increments while the omnidirectional transmitter remains fixed. The same procedure is repeated for each of 36 increments. Moving the transmitter changes not only the angle relative to the antenna but also the nodes' positions relative to their environment. To address this confound, each physical position is treated as a separate experiment. This means that the number of angles *relative to the steered antenna pattern* is limited to the number of distinct antenna states (16). The transmission power of the radio attached to the directional antenna was turned down to 10dBm to produce more tractable noise effects (the default EIRP is much too high to model small scale behavior).

Parabolic-Indoor-A and Patch-Indoor-A: This data set was collected in the University of Colorado Computer Science Systems Laboratory. The directional transmitter was positioned approximately 6 m from the receiver in a walkway between cubicles and desks. This is the most cluttered environment studied.

Parabolic-Indoor-B, Parabolic-Indoor-C, Patch-Indoor-B, and Patch-Indoor-C: An indoor office space was used for this set of tests. See figure A.3 for the floor-floorplan of this office space. Two receivers were used here: one with line of sight and one without line-of-sight, placed amidst desks and offices.

Array-Indoor-A and Array-Indoor-B: Seven phased array antennas are deployed in the same 25x30m indoor office space used for Parabolic-Indoor-B, Parabolic-Indoor-C, Patch-Indoor-B and Patch-Indoor-C. Data from two of the seven antennas are analyzed here. Each antenna electronically steers through its 16 directional states, spending 20 ms at each state. Two mobile omnidirectional transmitters move through the space and transmit 500 packets at 44 distinct positions. For each packet received by a phased array, the packet's transmission location and orientation is recorded (i.e., which of the four cardinal directions was the transmitter facing) along with the directional state in which the packet arrived and the RSSI value.

Parabolic-Reference and Patch-Reference: The large floodplain is used here. An Agilent VSA is connected to the omnidirectional receiver and makes a 10 second running average of power samples on a specific frequency (2.412 GHz was used). Three consecutive averages of both peak and band power are recorded for each direction. The directional transmitter is rotated in five degree increments and is connected to a VSG outputting a constant sinusoidal tone at 25 dBm on a specific frequency. Before, after, and between experiments, we make noise floor measurements, and as a postprocessing step, we subtract the mean of this value (-59.62 dBm or 1.1 nW) from the measurements.

A.3.2 Normalization

The task in comparing data sets is to come up with a scheme for normalization so that they can be compared to one another directly. For each data set, the mean peak value is determined, which is the maximum of the mean of samples for each discrete angle. This value is then subtracted from every value in the data set. The net effect is that the peak of the measurements in each data set will be shifted to zero, which allows comparison of measurements from diverse RF environments directly.

A.3.3 Error Relative to the Reference

Figure A.4 shows the normalized measured *in situ* patterns and their corresponding (also normalized) reference patterns. Recall that the reference pattern is generated and recorded by calibrated signal processing equipment and the measured data is collected using commodity 802.11 cards. There is much variation in

Label	Environment	LOS?	Dist. (m)	Samples	Loss (%)
Parabolic-Outdoor-A	Open Field on Campus	Yes	30.5	214471	24.81
Parabolic-Outdoor-B	Empty Floodplain	Yes	30.5	258876	7.05
Parabolic-Indoor-A	Laboratory	Yes	30.5	267092	2.21
Parabolic-Indoor-B	Office Building	Yes	≈ 60	268935	10.41
Parabolic-Indoor-C	Office Building	No	≈ 15	283104	5.12
Parabolic-Reference	Empty Floodplain	Yes	30.5	219	N/A
Patch-Outdoor-A	Open Field on Campus	Yes	30.5	455952	12.44
Patch-Outdoor-B	Empty Floodplain	Yes	30.5	278239	4.99
Patch-Indoor-A	Laboratory	Yes	30.5	290030	2.21
Patch-Indoor-B	Office Building	Yes	≈ 60	265593	7.40
Patch-Indoor-C	Office Building	No	≈ 15	278205	2.65
Patch-Reference	Empty Floodplain	Yes	30.5	219	N/A
Array-Outdoor-A	Open Field on Campus	Yes	≈ 30	475178	N/A
Array-Indoor-A	Office Building	Mixed	Varies	2672050	N/A
Array-Indoor-B	Office Building	Mixed	Varies	2708160	N/A
Array-Reference	Open Urban Area	Yes	≈ 5	360	N/A

Table A.1: Summary of data sets.

the measured patterns and in how much they differ from the reference (which would be typically classified as error). As might be expected, the measurements in outdoor environments exhibit less noise due to less clutter, but still deviate from the reference on occasion. As a further confirmation that the measurement process works well, notice how well Parabolic-Outdoor-B and Patch-Outdoor-B (figures A.4(b) and A.4(d)) correlate with the reference pattern (recall that these experiments were done in the same floodplain as the reference, indicating that the commodity hardware can compete with the expensive specialized equipment in a similar environment).

On inspection of this data, the first question is whether there a straightforward explanation for error in the measured patterns. Figure A.6 provides a CDF of all error for each antenna. The three antennas provide similar error distributions, although offset in the mean. The array data is the most offset from the others (presumably because its reference pattern is theoretical rather than measured) and exhibits some bimodal behavior. The patch measurements are closest to the reference, showing a large kurtosis about zero. Figure A.5 shows a PDF of error averaged at each angle—discarding outliers this way, the error between antennas begins to suggest similar distributions.

Clearly, the antennas have different error characteristics. However, for each antenna, and for each data set, it might be that the error in a given direction is correlated with that in other directions—if this were true, a single or small set of probability distributions could be used to describe the error process in a given environment with a given antenna.

A Shapiro-Wilkes test is used on the per angle error for each data set. The resulting p-values are well under the $\alpha = 0.05$ threshold, and in all cases the null hypothesis that the error is normally distributed can be rejected; this means that standard statistical tests (and regression models) that assume normality cannot be used. A pairwise Mann-Whitney U-test can be used to determine which pairs of samples grouped on some criterion (in this case, angle) are drawn from the same distribution. For each data set, a “heatmap” is generated where each cell corresponds to a pair of angles. The cell is colored by the p-value produced by the U-test when run pairwise, comparing the error for the reference pattern and the *in situ* pattern for those angles. Remarkably, all of the traces produce similar heatmaps: in the majority of pairs the null hypothesis that their error process is drawn from the same distribution can be rejected. However, for angles

near zero, this hypothesis cannot be rejected. This observation, that *measurements where the main lobe of the directional antenna is pointed at the receiver may exhibit correlated error processes*, motivated another series of tests.

To further explore "possibly well behaved" error processes about the main lobe, a Kruskal-Wallis rank-sum test was applied to two scenarios: (1) For angles near zero, are batches with the same antenna (but different environments) equivalent? (2) For angles near zero, are batches with the same environment (but different antennas) equivalent?

For (1), the null hypothesis is soundly rejected for all combinations (p-value $\ll 0.05$) For (2), the results still point strongly toward rejection (mean p-value = 0.0082), however there is one outlier—in the case of 355 degrees in the laboratory environment, a p-value of 0.2097 is achieved. One outlier, however, is not sufficient to overcome the evidence that neither antenna configuration nor environment alone is sufficient to account for intra-angle variation in error—even in the more seemingly well behaved cone of the antenna mainlobe.

A.3.4 Observations

There are several qualitative points that are worth bringing out of this data: (1) In the indoor environments, none of the measurements track the reference signal at all closely; (2) In all environments, there is significant variation between data sets; (3) The maximum signal strength is generally realized in *approximately* the direction of maximum antenna gain, but directions of low antenna gain often do not have correspondingly low signal strength. This means that *no system for interference mitigation can safely rely on predetermined antenna patterns*.

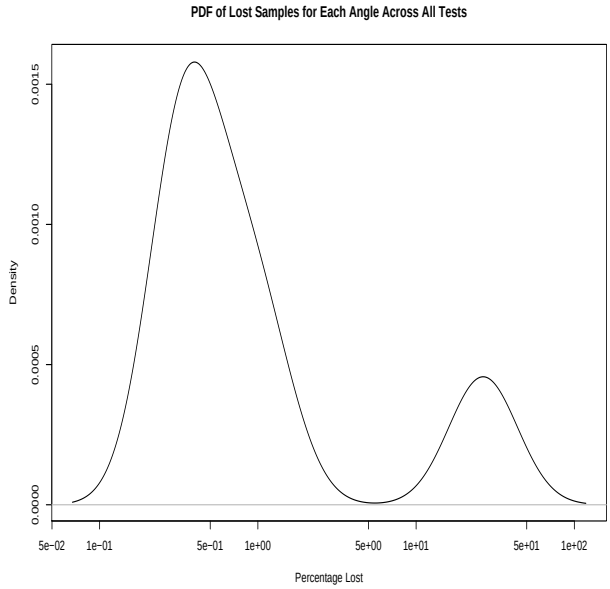


Figure A.1: Probability Density Function (PDF) of percentage of dropped measurement packets in a given angle for all angles and all data sets.

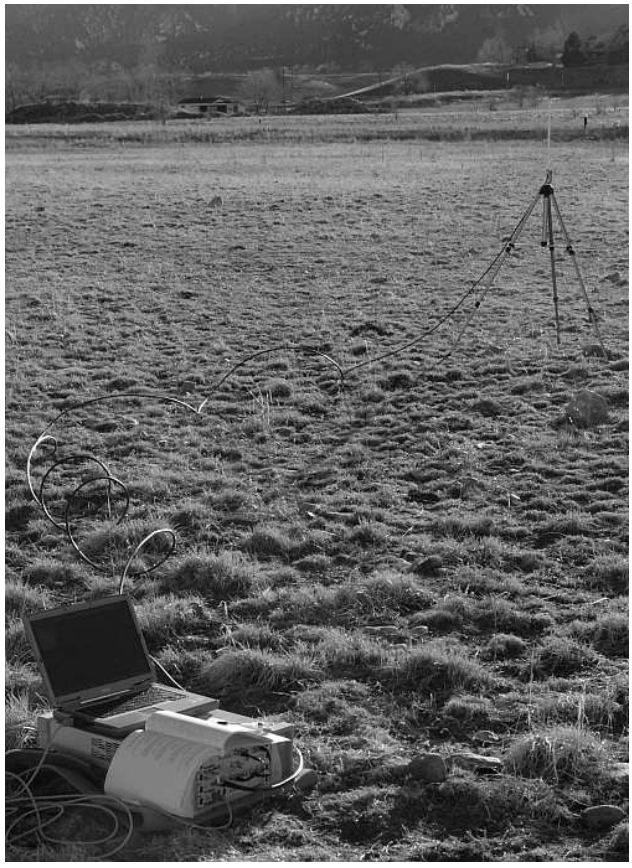
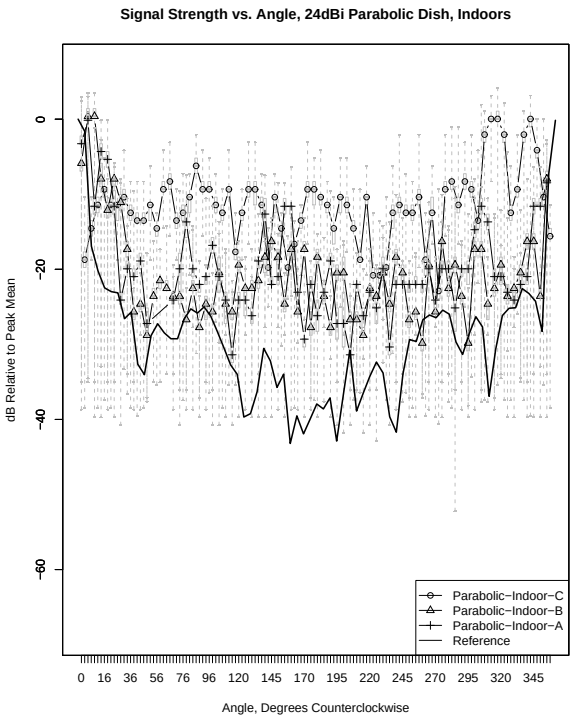
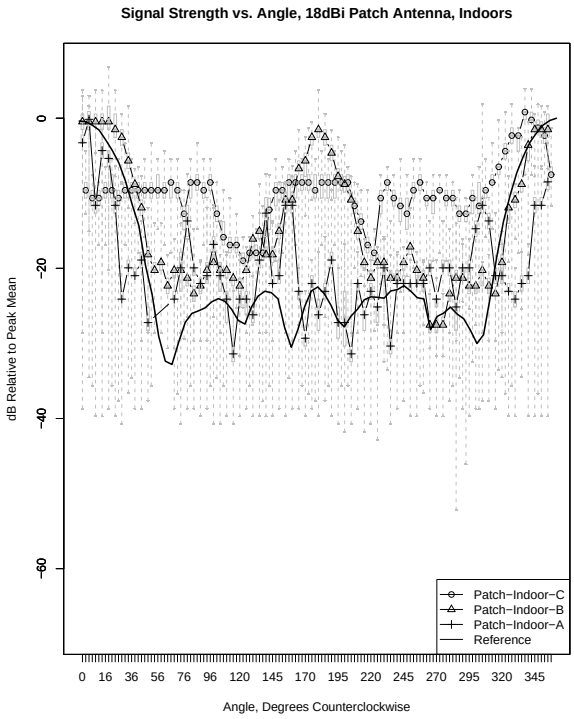


Figure A.2: Receiver side of measurement setup in floodplain

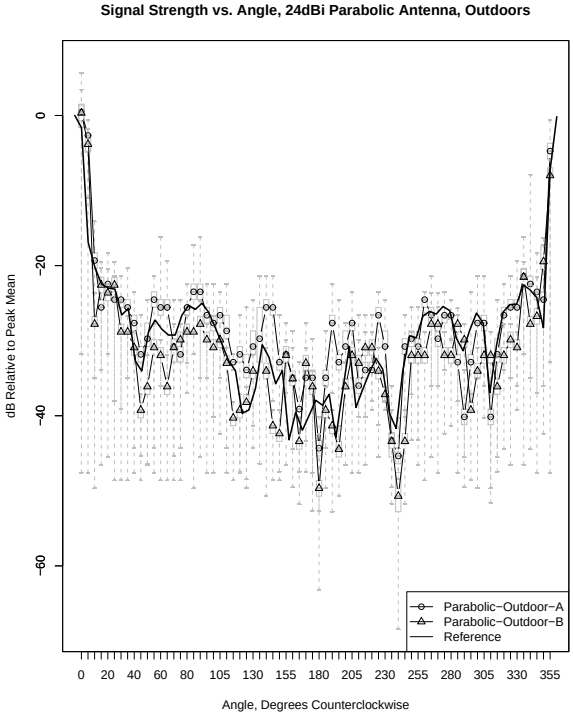
Figure A.3: Floorplan of office building used in Array-Indoor-A, Array-Indoor-B, Patch-Indoor-B, Patch-Indoor-C, Parabolic-Indoor-B, and Parabolic-IndoorC.



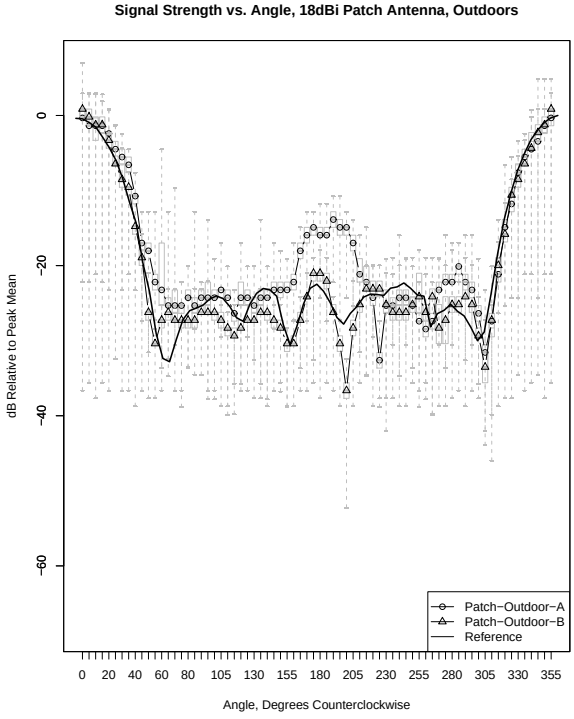
(a) Parabolic dish indoor environments



(c) Patch panel indoor environments



(b) Parabolic dish outdoor environments



(d) Patch panel outdoor environments

Figure A.4: Comparison of signal strength patterns across different environments and antennas.

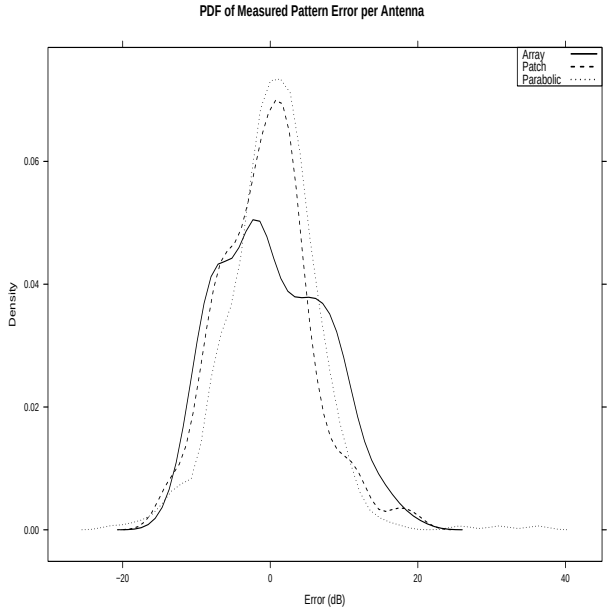


Figure A.5: Probability Density Functions (PDFs) for the averaged error process (combined across multiple traces) for each antenna type.

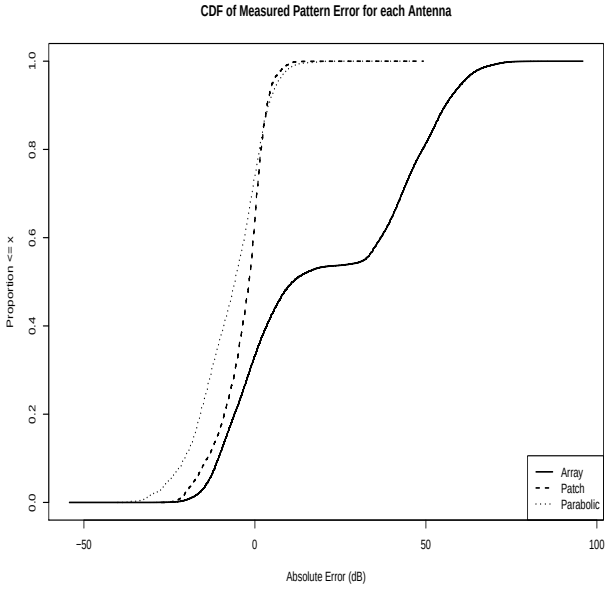


Figure A.6: Cumulative Density Functions (CDFs) for the error process (combined across multiple traces) for each antenna type.

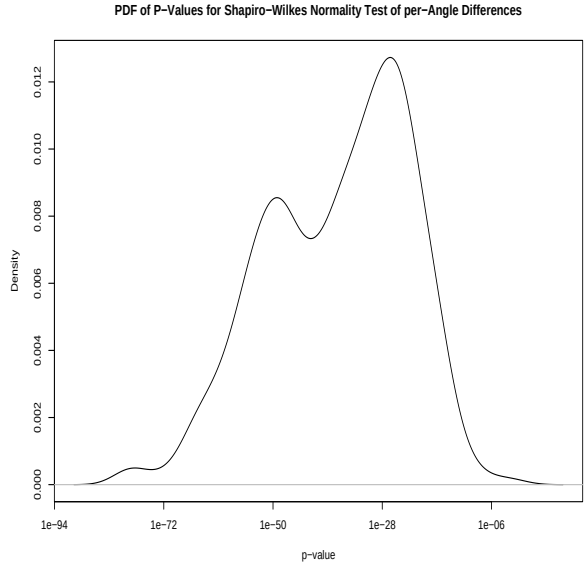


Figure A.7: Probability Density Function (PDF) of p-values from testing the normality of the error process in each direction for each data set. In all cases, the null hypothesis (that the samples are normally distributed) can be confidently rejected.

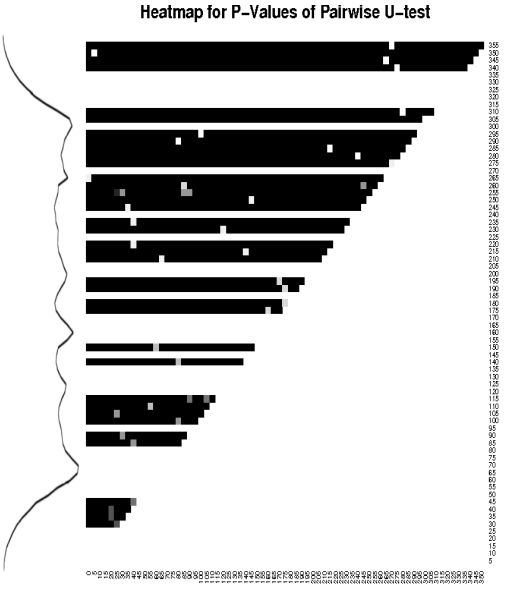


Figure A.8: Heatmap of p-values for the Mann-Whitney U-test which was run pairwise against the error from the reference pattern in each angle. This plot, which is for Patch-Indoor-A, was chosen as a representative. *All traces showed similar trends.* Darker values indicate very small p-values, meaning that the null hypothesis can be rejected with confidence. In this case, the null hypothesis is that the samples come from the same distribution. The Patch reference pattern is provided on the left for reference.

A.4 A New Model of Directionality

This chapter began with the observation that path loss and antenna gain are typically regarded as orthogonal components of the power loss between transmission and reception (equations A.1 – A.3). In this section, the *best case* accuracy of this approach is evaluated, and a new model based on the limitations identified is derived.

A.4.1 Limitations of Orthogonal Models

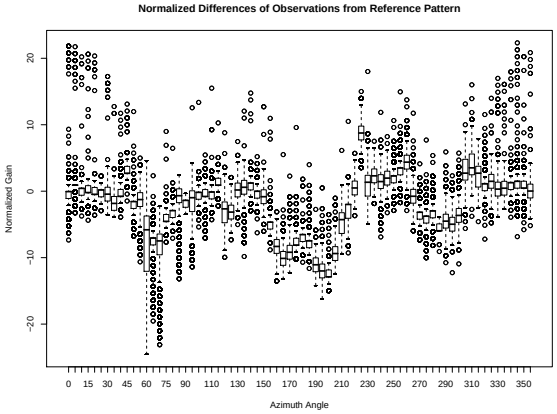
If transmit power and path loss do not vary with antenna angle, the received power relative to antenna angle can be modeled as:

$$\widehat{P_{rx}} = \beta_0 * f(\phi, \theta) \tag{A.6}$$

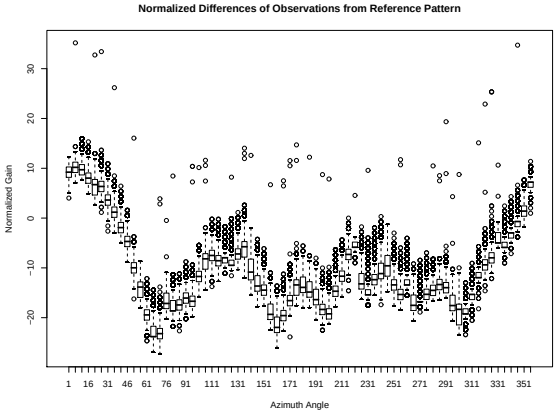
β_0 is a constant combining the path loss—however calculated—and the gain of the nonrotating antenna. $f(\phi, \theta)$ is a function describing the gain of the other antenna relative to the signal azimuth θ and zenith ϕ. Without loss of generality, an assumption is made that the antenna being varied is the receiver, and that the zenith, ϕ, is fixed.

To evaluate the accuracy of this model, the estimate b_0 for β_0 is found that minimizes the sum of squared error (SSE). In effect, this is assuming the *best possible path loss estimate*, without specifying how it is determined. If the function $f(\phi, \theta)$ correctly describes the antenna, and if path loss and antenna gain are in fact orthogonal components of the received signal strength, then the remaining error should be *randomly* distributed about 0.

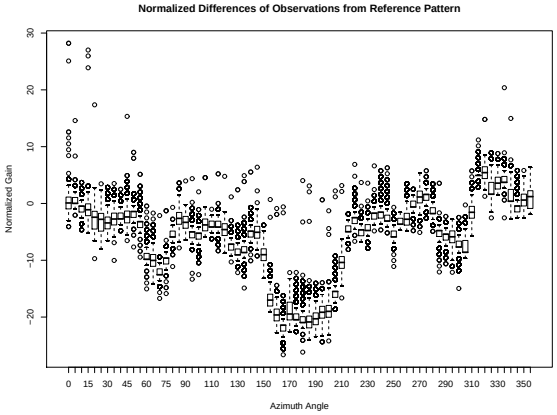
Figure A.9 depicts the error of this *orthogonal model* for several data sets. There are several qualitative observations to be made: First and most importantly, *the error value is not uniformly random, but rather correlated with direction.* The variability within any given direction is less than that for the data set as a whole. Second, the error is significant. In the worst states, the *mean error* is between 8 and 10 dB, in either direction. Third, the model overestimates signal strength in the directions where the gain is highest, and underestimates in the directions where the gain is lowest. That is, *the difference in actual signal strength between peaks and nulls is less than the antenna in isolation would produce.* This has significant



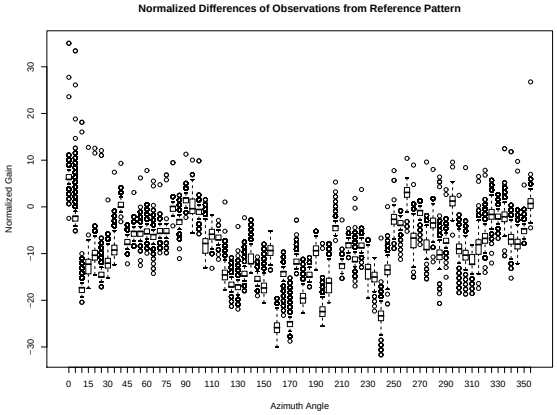
(a) Patch-Outdoor-A



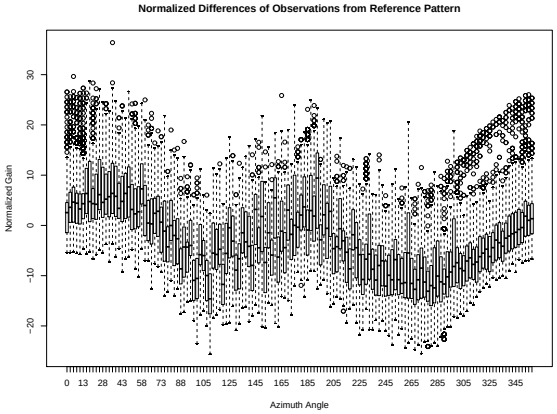
(b) Patch-Indoor-B



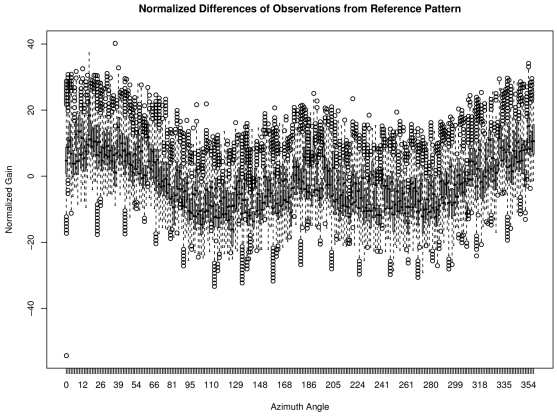
(c) Patch-Indoor-C



(d) Parabolic-Indoor-C



(e) Array-Outdoor-A



(f) Array-Indoor-A

Figure A.9: Differences between the orthogonal model and observed data in dB: $\hat{P_{rx}} - P_{rx}$.

implications for systems that use null steering to manage interference.

The data in figures A.9(e) and A.9(f) is aggregated from 36 distinct physical configurations. In each configuration, the directional receiver was (electronically) rotated in 22.5 degree increments, and between configurations, the omnidirectional transmitter was physically moved around the receiver by ten degrees. A consequence of this method is that these 10 degree changes represent not only a change of the angle between the transmitter and the antenna, but also a change of location with the attendant possibility of fading effects. To account for this, each of the 36 configurations is considered individually. This gives less angular resolution, but also fewer confounds. Figure A.10 displays each configuration as a separate line. The model accuracy is fairly consistent: The residual standard error of the aggregate is 8 dB, and the individual cases range from 5.74 dB to 11.4 dB with a mean of 7.6 dB.

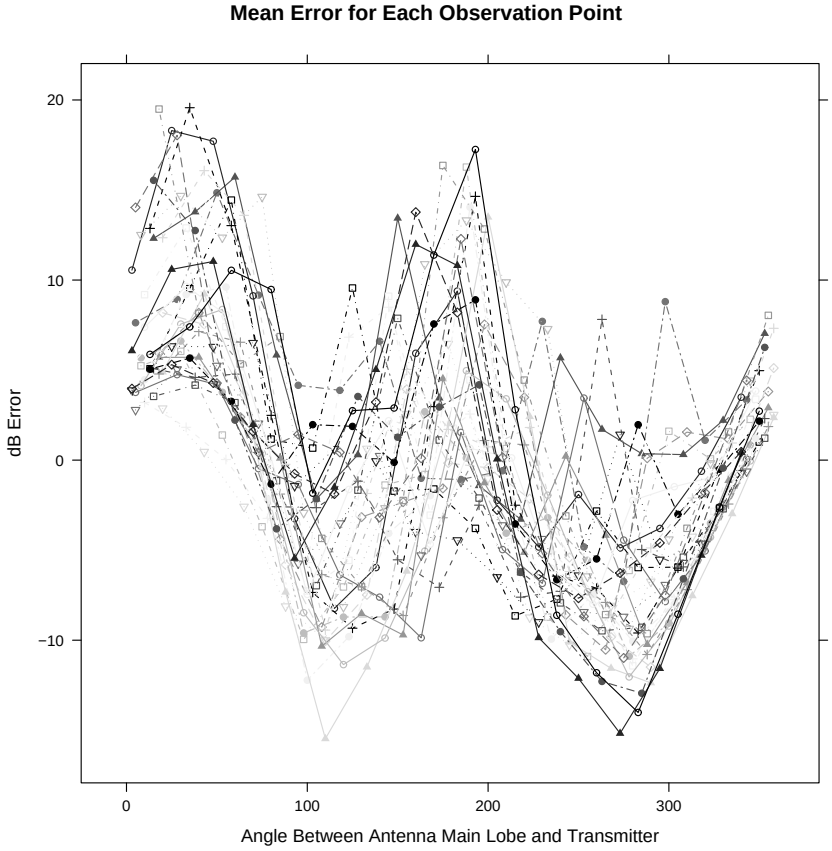


Figure A.10: Mean error of orthogonal model for each observation point in the Array-Outdoor-A data set. The format is the same as in figure A.9.

The path loss value used for each data set was the lowest error fit for that specific data, and the antenna patterns ($f(\theta)$) for the patch and parabolic antennas were measured using the specific individual antenna in question. Note also that error patterns differ from environment to environment: one could derive an *ex post facto* $f(\theta)$ to eliminate the error in a single data set, but it would not be applicable to any other.

The magnitude and *systematic nature* of the error suggest that the orthogonal model has inherent limitations that cannot be alleviated by improving either the antenna model or path loss model separately.

A.4.2 An Integrated Model

To address these limitations, an integrated model is derived that addresses the systematic errors discussed above, while remaining simple enough to use in analysis and simulations.

The *environment specific, direction specific* error shown in figure A.9 is addressed with the following environment aware model, given in equation A.7. The expected received power is given by a constant β_0, the antenna gain function $f(\phi, \theta)$, and a yet to be determined environmental offset function $x(\phi, \theta)$:

$$\widehat{P_{rx}} = \beta_0 * f(\phi, \theta) * x(\phi, \theta) \tag{A.7}$$

As with the orthogonal model, a constant zenith is assumed and $f(\phi, \theta)$ and $x(\phi, \theta)$ are considered with regard to the azimuth θ. Equation A.7 can be converted to a form that lends itself to least squares (linear regression) analysis in the following way: First, equation A.7 is rewritten as addition in a logarithmic domain, and second a discrete version of the general $x(\theta)$ is substituted in. In the discrete $x(\theta)$, the range of angles is partitioned into n bins such that bin i spans the range $[B_i, T_i)$. Each bin has associated with it a boxcar function $d_i(\theta)$ to be 1 if and only if the angle θ falls within bin i (equation A.8) and an unknown constant *offset value* β_i. These transformations yield the model given in equation A.10.

$$d_i(\theta) = \begin{cases} 1, & B_i \leq \theta < T_i \\ 0, & \text{otherwise} \end{cases} \tag{A.8}$$

$$x(\theta) = \sum_{i=1}^{n} d_i(\theta)\, \beta_i \tag{A.9}$$

$$f(\theta) - \widehat{P_{rx}} = \beta_0 + \beta_1 d_1(\theta) + \beta_2 d_2(\theta) + \cdots + \beta_n d_n(\theta) \tag{A.10}$$

If $x(\theta)$ is discretized into n bins, the model has $n + 1$ degrees of freedom: One for each bin and one for β_0, the signal strength without antenna gain. For any given signal direction, exactly one of the $d_i(\theta)$ functions will be 1, so each prediction is an interaction of two coefficients: β_0 and β_i. Consequently, β_0 could be eliminated and an equivalent model achieved by adding β_0's value to each β_i. Mathematically, this means that there are only n independent variables in the SSE fitting, and the full set is collinear. In practice, the constant β_n is dropped, but this does not mean that packets arriving in that bin are any less well modeled. Rather, one can think of bin n as being the "default" case.

The azimuth can be divided into arbitrarily many bins. The more finely it is divided, the more degrees of freedom the model offers, and thus the more closely it can be fitted to the environment. To investigate the effect of bin number, every data set is modeled using from two to twenty bins. Figure A.11 shows the residual standard error as a function of bin count. The grey box plot depicts the mean and interquartile range for all of the data collectively, and the foreground lines show values for links individually. In general, there appears to be a diminishing return as the number of bins increases, with the mean remaining nearly constant above 16 bins.

In discussing parameters for this model, we will use the 16-bin case specifically. We find the same patterns across other numbers, though the actual coefficients are bin count specific. One result of note with regard to bin count is this: Several environments exhibit a "sawtooth" pattern in which the odd bin counts do better than the even ones, or vice versa. This appears to be an effect of the *alignment* of the bins relative to environmental features, rather than the *number* of bins as such.

The model described here has significantly less error than the orthogonal model: Across all data sets, the mean residual standard error is 4.0 dB, *(4.4dB indoors)* compared to 6.15 dB *(7.312 dB indoors)* for the orthogonal model. More importantly, the error remaining in the discrete offset model is largely noise: The mean error is almost exactly zero for several ways of grouping the data. Figure A.12 depicts the error (predicted value minus observed value). While the outliers reveal some direction correlated effect that is not accounted for, this model is much better for the bulk of the traffic. Over 99.9% of the traffic *at every angle* falls within the whisker interval.

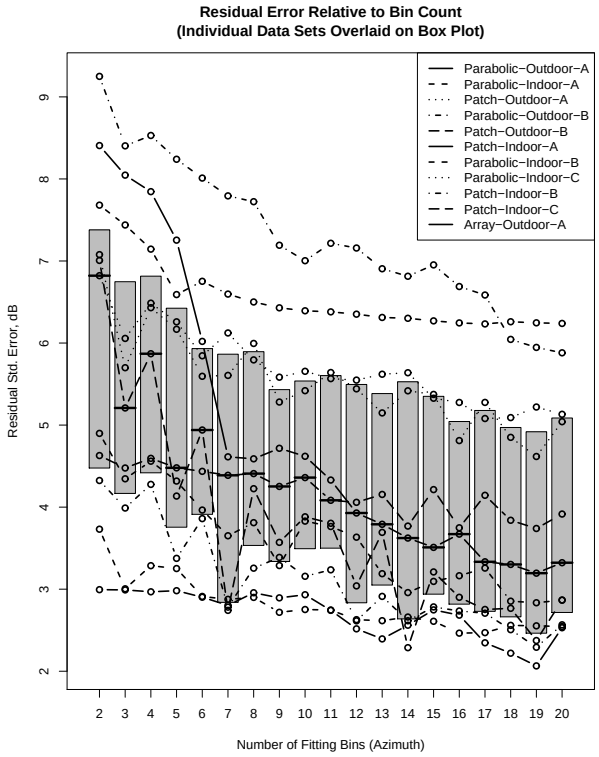


Figure A.11: Effect of increasing bin count (decreasing bin size) on modeling precision.

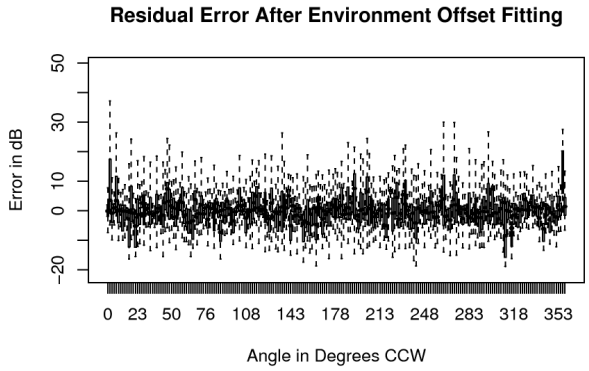


Figure A.12: Residual error of the discrete offset model with 16 bins.

A.4.3 Describing and Predicting Environments

The environmental offset function $x(\phi, \theta)$, or its bin offset counterpart, models the impact of a particular environment combined with a particular antenna. This can serve as an *ex post facto* description of the environment encountered, but it also has predictive value: If one knows the offset function for a given environment, it is possible to more accurately model wireless systems in that environment. There is no practical way to know the exact spatial RF characteristics of an environment—and thus its offsets—without actually measuring it. However, these results suggest that it is possible to identify parameters generating the *distribution* from which the offset values for a *class of environments* are drawn.

An analysis of possible determining factors for the fitted offsets was conducted across all traces and a range of bin counts. A linear regression fit and ANOVA test found significant correlation with two factors: The nominal antenna gain $f(\theta)$ and the observation point; none of the other factors examined were consistently significant. Table A.2 shows the regression coefficients and P-values for both factors for a variety of traces. The observation angle was always statistically significant, but the coefficient is constantly near zero. For each factor, the regression coefficient describes the correlation between the fitted offset and the factor. That is, the coefficient shows how much the actual signal strength can be expected to differ from the orthogonal model, for any value of that factor. For example, the antenna gain coefficients of .668 and .703 for Parabolic-Indoor-C and Patch-Indoor-C mean that in those data sets for every dB difference in antenna gain between two angles, *the best fit difference in actual signal strength is only* $\approx$ *0.3 dB.*

There are two key results pertaining to the antenna gain regression coefficient: First, the coefficients for different antennas in the same environment are very close. Second, the coefficients for distinct but similar environments are fairly close. This suggests that classes of environments can reasonably be characterized by their associated coefficients, which provides a compact representation of environment classes that lends itself easily to simulation. In this way, the task of the researcher is reduced to choosing amongst several *representative environment classes* when designing their experiment.

Data Set	Factor	Coefficient	P-value
Parabolic-Outdoor-A	Antenna Gain	0.185	1.02e-87
	Obs. Angle	0.00301	5.1e-06
Patch-Outdoor-A	Antenna Gain	0.146	6.4e-50
	Obs. Angle	0.00744	1.14e-17
Array-Outdoor-A	Antenna Gain	0.41	2.03e-206
	Obs. Angle	-0.0271	5.36e-188
Parabolic-Outdoor-B	Antenna Gain	0.0377	8.68e-05
	Obs. Angle	-0.00323	5.95e-05
Patch-Outdoor-B	Antenna Gain	0.00919	0.0492
	Obs. Angle	-0.00198	3.08e-06
Parabolic-Indoor-A	Antenna Gain	0.33	4.6e-102
	Obs. Angle	0.00463	1.91e-05
Patch-Indoor-A	Antenna Gain	0.258	1.22e-122
	Obs. Angle	0.00894	3.09e-24
Parabolic-Indoor-B	Antenna Gain	0.378	2.2e-134
	Obs. Angle	0.00971	1.97e-16
Patch-Indoor-B	Antenna Gain	0.372	1.1e-81
	Obs. Angle	0.014	3.87e-18
Parabolic-Indoor-C	Antenna Gain	0.668	1.39e-234
	Obs. Angle	-0.0146	4.15e-36
Patch-Indoor-C	Antenna Gain	0.703	0
	Obs. Angle	-0.0154	2.63e-48

Table A.2: Factors influencing fitted offset values, 16-bin case.

A.5 Simulation Process

The statistical model laid out above can be used as the basis for more realistic simulations. It has long been recognized that radio propagation involves very environment specific effects. Three major ways of addressing such effects in modeling and simulation have been identified: The first is to simply ignore the variability and use a single representative value in all cases. The second, which goes to the opposite extreme, is to model specific environments in great detail. A third approach is to randomly generate values according to a representative process and perform repeated experiments.

Each approach has its benefits, but this chapter advocates the repeated sample approach. Precisely modeling a specific environment probably has the greatest fidelity, but it provides no information as to how well results achieved in a single environment will generalize to others. Stochastic models have the advantage of being able to produce arbitrarily many "similar" instances, and parametric models make it possible to study the impact of varying a given attribute of the environment. Such approaches are frequently used to model channel conditions [151], network topology [247, 218], and traffic load [129].

The following algorithms produce signal strength values consistent with our statistical findings. The key parameters are the gain offset correlation coefficient K_{gain}, the offset residual error S_{off}, and the per packet signal strength residual error S_{ss}. These values were computed across many links for two types of environments in sections A.4.3 and A.4.2. Table A.3 summarizes these results.

Environment	K_{gain}	S_{off}	S_{ss}
Open Outdoor	0.01 - 0.04	1.326 - 2.675	2.68 - 3.75
Urban Outdoor	0.15 - 0.19	2.244 - 3.023	2.46 - 2.75
LOS Indoor	0.25 - 0.38	2.837 - 5.242	2.9 - 5.28
NLOS Indoor	0.67 - 0.70	3.17 - 3.566	3.67 - 6.69

Table A.3: Summary of Data Derived Simulation Parameters: Gain-offset regression coefficient (K_{gain}), offset residual std. error (S_{off}), and signal strength residual std. error (S_{ss}).

Algorithm 5 is a one time initialization procedure which computes the offsets between the antenna gain in any direction and the expected actual signal gain.

Algorithm 6 computes the expected end to end gain for a given packet, *not including fixed path loss*. Thus, the simulated signal strength would be determined by the transmit power, path loss, receiver gain,

Algorithm 5 Compute direction gain

1: $K_{gain} \leftarrow$ gain offset correlation coefficient
2: $S_{off} \leftarrow$ offset residual std. error
3: **procedure** DIRECT-GAIN
4: **for** Node $n \in$ all nodes **do**
5: $P \leftarrow$ partition of azimuth range $[-\pi, \pi)$
6: **for** $p_i \in P$ **do**
7: $\theta_i \leftarrow$ center angle of p_i
8: $X \leftarrow$ random value from $(\mu = 0, \sigma^2 = S_{off})$
9: $o_{n,p_i} \leftarrow K_{gain} * f_n(\theta_i) + X$
10: **end for**
11: **end for**
12: **end procedure**

fading model (if any) and the directional gain from algorithm 6. Note that a fading model that accounts for interpacket variation for stationary nodes might make the random error ϵ in line 9 redundant.

A.6 Summary and Conclusion

This chapter has presented an empirical study of the way different environments and antennas interact to affect the directionality of signal propagation. The three primary contributions of this work are:

(1) A well validated method for surveying propagation environments with inexpensive commodity hardware.

(2) A characterization of several specific environments ranging from the very cluttered to the very open.

(3) New, more accurate, techniques for modeling and simulating directional wireless networking.

Wireless signal—and interference—propagation is more complicated than common previous models have acknowledged. Because models of the physical layer guide the development and evaluation of higher layer systems, it is important that these models describe reality well enough. Indeed, [30] shows that application layer results reported by simulators can be affected dramatically by the way directional antenna models are simulated, producing results that deviate significantly from reality. The measurements described here, and the resulting model, bring to light several important aspects of the physical environment that previous models have failed to capture. The *effective* directionality of a system depends not only on the antenna,

Algorithm 6 Compute per-packet gain

$S_{pss} \leftarrow$ residual error of packet signal strengths
function DIRECTIONAL-PACKET-GAIN(src, dst)
    $\theta_{src} \leftarrow$ direction from src toward dst
    $\theta_{dst} \leftarrow$ direction from dst toward src
    $p_{src} \leftarrow$ partition at src containing θ_{src}
    $p_{dst} \leftarrow$ partition at dst containing θ_{dst}
    $G_{src} \leftarrow f_{src}(\theta_{src}) - o_{src,p_{src}}$
    $G_{dst} \leftarrow f_{dst}(\theta_{dst}) - o_{src,p_{dst}}$
    $\epsilon \leftarrow$ random value from $(\mu = 0, \sigma^2 = S_{pss})$
    return($G_{src} + G_{dst} + \epsilon$)
end function

but is influenced by the environment to such a large extent that many decisions cannot be made without *in situ* measurements.

Appendix B

The Stability of The ITM for Typical Problems

The ITM is a well known and widely used model for predicting propagation loss in long (greater than one kilometer) outdoor radio links. This model was developed by Hufford *et al.* in [99] for the National Telecommunications and Information Administration (NTIA) Institute for Telecommunications Sciences (ITS). The model predicts the median attenuation of the radio signal as a function of distance and of losses due to refractions at intermediate obstacles. Compared to the vast majority of other models, even those that are similar in approach (e.g., The International Telecommunications Union (ITU) Terrain Model [206]), the ITM is very complicated, requiring the interaction of dozens of functions that implement numerical approximations to theory. Due to this complexity, the question of numerical stability is an obvious one, but has not previously been investigated.

This section takes a systematic empirical approach to the analysis that involves porting the defacto C++ implementation of the ITM [98] to a multiprecision framework. A comparison is made between the predicted path loss values for many randomly generated links over real terrain data. Model parameters are also varied in order to produce a fully factorial experimental design over a range of realistic parameters. In the end, the results show that while the model performs disastrously for half-precision (16 bit) arithmetic, it is well behaved for single-precision (64 bit) and higher precisions. Within the values tested, there are very few isolated cases that result in significantly different (greater than 3 dB) output and these tend to result from a single change in branching decision in the approximation algorithms and not because of massive information loss. While this sort if empirical analysis cannot be used to extrapolate to any parameters and

[0] Work in this appendix has appeared in [173].

any terrain model, the results show that over realistic links the model appears to be well-behaved. This result provides confidence in the stability of the output of the ITM model as well as other similar models that compute diffraction over terrain (e.g, [206, 109]).

B.1 Implementation

The multiprecision framework used here is based on the combination of three open source libraries: MPL, MPC, and MPFR [10, 11, 7]. The MPL library provides basic arbitrary precision support. The MPFR library wraps the MPL library and provides additional necessary features such as a square root function, computation of logs and powers, and trigonometric functions. The MPC library provides support for complex arithmetic. In porting, the ITM source is modified to take an additional command line argument that specifies the precision in bits, which is passed to the multiprecision framework. Otherwise, the functionality and usage is identical to the machine-precision ITM implementation.

The implementation involves a line by line port of the reference ITM implementation to have multiprecision support. By and large, this involves using multiprecision data structures in place of native machine number formats. For instance, The following (commented) equation might be translated into four MPFR function calls:

```
1  #fhtv=0.05751*x-4.343*log(x);
2  mpfr_log(tmp,x,R);
3  mpfr_mul_d(tmp,tmp,4.343,R);
4  mpfr_mul_d(fhtv,x,0.05751,R);
5  mpfr_sub(fhtv,fhtv,tmp,R);
```

B.2 Experiment

The experimental design involves generating random link geometries within a latitude and longitude bounding box. For each random link, a path loss prediction is made both with the machine precision (64-bit double precision arithmetic) and multiprecision implementation (at a variety of precisions). After the fact, we can quantify the differences in predictions and investigate any outliers or general trends.

The bounding box is from 39.95324 to 40.07186 latitude and -105.31843 to -105.18602 longitude. This box contains a portion of the mountainous region to the west of Boulder, Colorado, as well as the plains

to the east, providing a realistic mix of topographies. 500 links are generated uniformly at random within the box. Antenna heights are also selected uniformly at random between 0 and 35 meters. For each link, the corresponding elevation profile is extracted from a USGS DEM with 0.3 arcsecond raster precision.

B.3 Results

Figure B.1 shows the overall results of this experiment: the error (ϵ) between the multiprecision prediction and the machine precision prediction is plotted. Half-precision arithmetic (11 bits of exponent, 16 bits total) produces results that vary wildly. Above this, however, beginning at single precision (24 bits of exponent, 32 bits total), the two programs make very similar predictions. Figure B.2 provides a more detailed picture of these remaining cases. Much of the small error is negligible as it is presumably a function of differences in rounding[1]. In the results, there is one clear outlier that produces a 6 dB difference. The case was the result of a difference in branching decision that chooses whether or not to make a correction. It is not clear that one direction down the branch offers a better prediction than another, so this case can be safely ignored.

Lastly, figure B.3 shows the performance, in terms of running time for the various precisions. The multiprecision version is not substantially slower than the machine precision implementation. If it were the case that the multiple precision implementation was also safer, then its use would be clearly preferable.

B.4 Discussion

Although it is not possible to extrapolate universally from these results, they demonstrate that the ITM is *not* substantially unstable for typical problems and reasonably precise numeric types (i.e., single and double precision IEEE formats). An analytical investigation of stability would go a long way to determine the stability universally, but is a substantial undertaking that involves the careful dissection of dozens of complex algorithms that combine to create the ITM implementation. An intrepid investigator, may choose to focus his effort on the knife-edge diffraction approximation algorithm, which is almost certainly the most

[1] IEEE 754-2008 requires subnormal arithmetic rounding, which is not done natively by the MPFR library. The majority of rounding (excluding this special case) are identical

numerically complex component of the model. For our purposes, however, the results presented here are sufficient to justify continued use of this model with the confidence that under typical situations it is not significantly affected by rounding and cancellation errors.

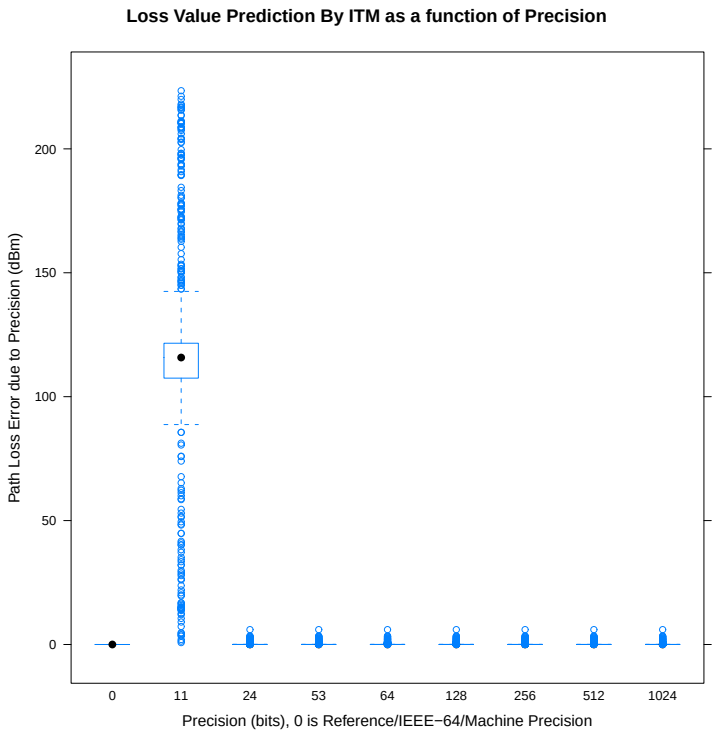


Figure B.1: Box and whiskers plot of error as a function of precision.

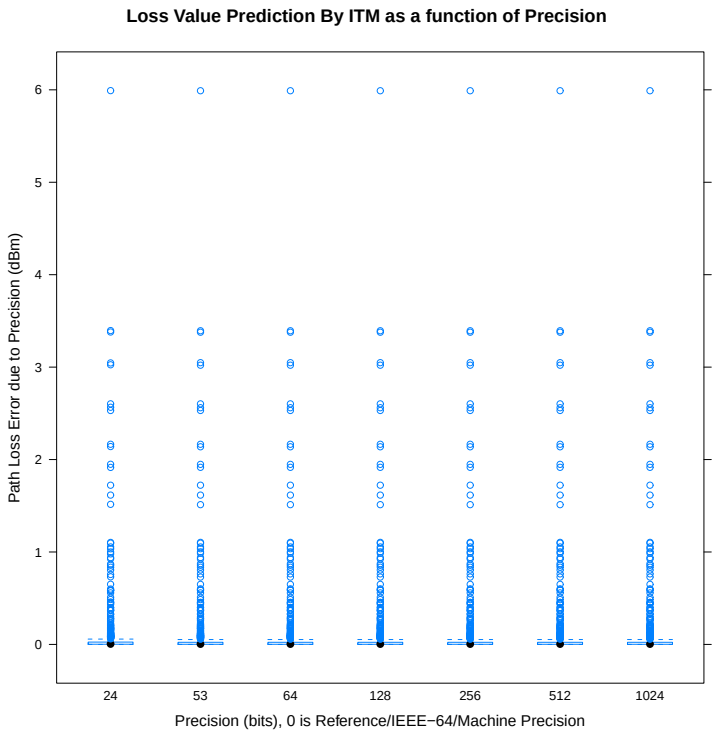


Figure B.2: Box and whiskers plot of error as a function of precision, showing only results for single-precision and greater arithmetic.

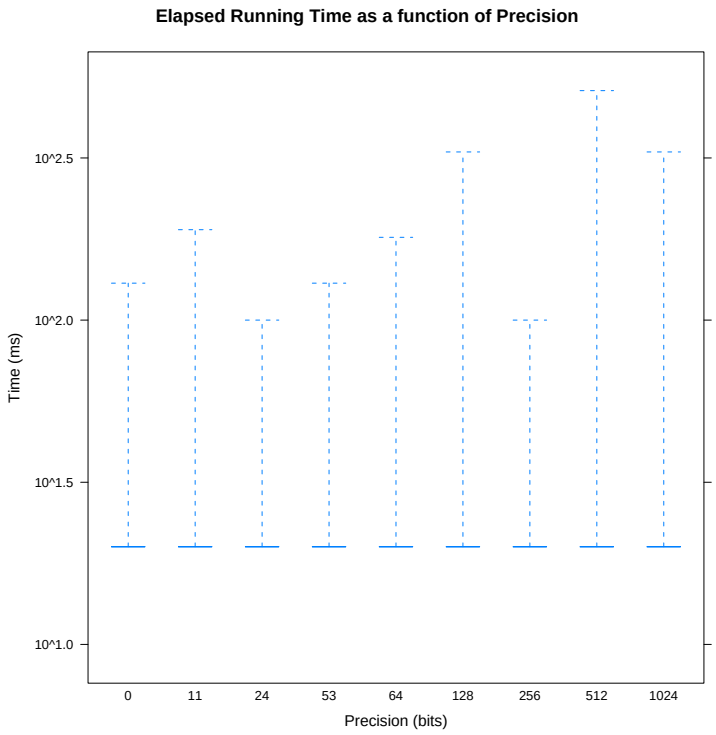


Figure B.3: Running time of ITM algorithm as a function of precision. The 0-bit case is the machine-precision reference implementation.

Appendix C

Reference Source Code

C.1 Kriging Variance

The implementation of kriging variance computation (equation 8.1) used in this thesis, in the R statistical computing language, using the geoR library is:

```
# This definition of kriging variance taken from
# equation (1) of Delmelle \textit{et al.} (2009), and
# is compatable with equation 4.18 in Spatial Statistics by
# Ripley. It appears to be a constant shift off the kriging
# variance computed by the geoR krige.conv method
krige.var <- function(dcoords,loci,kc){
  cs <- cov.spatial(obj=loccoords(coords=dcoords,locations=loci),
        cov.model=kc$cov.model,cov.pars=kc$cov.pars,
        kappa=kc$cov.kappa)
  vcinv <- varcov.spatial(coords=dcoords, cov.model=kc$cov.model,
           cov.pars=kc$cov.pars,kappa=kc$kappa,nugget=kc$nugget,
           inv=TRUE)$inverse
  sigmak <- NULL;
  sigmasq <- kc$cov.pars[1]
  for(i in seq(1,nrow(loci))){
    v <- sigmasq - t(cs[,i]) %*% vcinv %*% t(t(cs[,i]))
    sigmak <- rbind(sigmak,v)
  }
  rm(vcinv,cs)
  return(sigmak)
}
```

The for-loop in this function involves the bulk of computation, but can be parallelized like so:

```
# argument 4 is a 'cluster' made with a command like makeForkCluster(N)
krige.var.par <- function(dcoords,loci,kc,c1){
  cs <- cov.spatial(obj=loccoords(coords=dcoords,locations=loci),
                    cov.model=kc$cov.model,cov.pars=kc$cov.pars,
                    kappa=kc$cov.kappa)
  vcinv <- varcov.spatial(coords=dcoords, cov.model=kc$cov.model,
                          cov.pars=kc$cov.pars,kappa=kc$kappa,
                          nugget=kc$nugget,inv=TRUE)$inverse
  sigmasq <- kc$cov.pars[1]
  sigmak <- parLapply(c1,seq(1,nrow(loci)),
                      function(i){ sigmasq - t(cs[,i]) %*%
                                   vcinv %*% t(t(cs[,i])) })
  rm(vcinv,cs)
  return(as.numeric(sigmak))
}
```

C.2 Path Loss Prediction

The following code provides a ruby class that models a “path”, and provides implementations (or wrappers for those with outside/reference implenentations) of the path loss models studied in chapter 3. As it is defined here, a path must be at least two points (transmitter and receiver), although the terrain models will need the path to include a number of intermediary points and their elevations. In order to conserve space only those models with substantial complexity have been included.

```
1  class Path
2
3    ############### HELPER FUNCTIONS ########################
4
5    # the angle, in degrees between this site and a given site, in the zenith
6    # i.e., the angle of the LOS path from the perspective of the transmitter
7    # if you want instead, the angle between the street and the line of sight path
8    # (i.e.,the angle from the receiver's perspective), then this is just the
9    # ascension negated (they are alternate internal angles, which by definition
10   # are congruent)
11   def ascension(x,i)
12     h1 = x.z
13     h2 = i.z
14     dh = h1 - h2
15     d = distance(x,i)
16     -(rad_to_deg(atan(dh/d)))
17   end
18
19   # path LOS bit-vector calculation as in splat.cpp:PlotPath()
20   #
21   # returns a path-sized array where the i^th element is true if there's
22   # no obstruction (i.e., los) and false otherwise. the 0^th element
23   # is nil since it is meaningless (los from transmitter to itself)
24   #
25   # splat.cpp does some cos() comparison voodoo I don't understand.
26   # here I'm just using plain-old right-triangle trigonometry.
27   # which is probably slow, but probably correct.
28   #
29   # path is an array of Site objects. the first is the transmitter.
30   # fresnel is the fraction of the fresnel zone that can be obscured before
31   #         we deem the path as non-los. nil means don't bother thinking about
32   #         fresnel (same as fresnel = 0.5 AFAICT)
33   # frequency in Mhz
34   def los(f,fresnel)
35     tx = self.tx
36     agl_tx = tx.z                # in meters
37     ret = Array.new(self.length+1,nil)
38     w = freq_to_wavelength(f/1000)       # in meters
39
40     return ret if tx.ele.nil?
41
42     (1..self.length-1).each{ |i|
43
44       rx = self[i]
45
46       next if rx.ele.nil?
47
48       d = distance(rx,tx)*1000           # in meters
49       agl_rx = rx.ele+rx.h               # in meters
```

```
50        x = agl_tx - agl_rx               # negative if rx is above tx
51        alpha = acos(x/sqrt(x**2 + d**2)) # angle from rx to tx in radians
52
53        los = true
54
55        if i > 1
56          # check to see if any point between this rx and the tx is in the way
57          (1..i-1).each{ |j|
58            int = self[j]
59            next if int.ele.nil?
60            di = distance(int,tx)*1000      # in meters
61            agl_int = int.ele+int.h         # in meters
62            xi = agl_tx - agl_int           # in meters
63
64            unless fresnel.nil?
65              # a whole bunch of calculations...
66              d_a = distance(int,rx)*1000                    # meters
67              d_b = distance(int,tx)*1000                    # meters
68              # length of line from rx to tx
69              r_ab = sqrt((d_a+d_b)**2 + (agl_tx-agl_rx)**2)
70              # angle from rx to tx in radians (always pos)
71              angle_ab = acos((d_a+d_b) / r_ab)
72              # angle should be neg if rx is higher than tx
73              angle_ab = angle_ab * -1 if agl_tx < agl_rx
74              # radius of fresnel lens at int
75              r_f = sqrt((1*w*d_a*d_b) / (d_a+d_b))
76              # distance from rx to los point above/below int
77              r_los = d_a/cos(angle_ab)
78              # "width" of fresnel lens from receiver's view
79              angle_f = acos(r_los/sqrt(r_f**2 + r_los**2))
80              # length of line from rx to int
81              r_ai = sqrt((agl_int-agl_rx)**2 + d_a**2)
82              # angle from rx to int (always pos)
83              angle_ai = acos(d_a/r_ai)
84              # angle should be neg if rx is higher than int
85              angle_ai = angle_ai * -1 if agl_int < agl_rx
86
87              # proceed in calculating fraction
88              f = nil
89              # zone isn't obscured at all
90              if angle_ai < (angle_ab - angle_f)
91                f = 0.0
92              # zone is totally obscured
93              elsif angle_ai > (angle_ab + angle_f)
94                f = 1.0
95              # if the zone is partially obscured...
96              else
97                # if we're exactly at the LOS center line
98                if angle_ai == angle_ab
99                  f = 0.5
100               elsif angle_ai > angle_ab
101                 # if we're below the LOS center line
102                 f = 0.5 - ((angle_ai-angle_ab) / (2*angle_f))
103               else
104                 # if we're above the LOS center line
105                 f = 0.5 + ((angle_ab-angle_ai) / (2*angle_f))
106               end
107               # need to invert if we're working with neg angles
108               f = 1 - f if angle_ab < 0
109             end
110
111             los = false if f > fresnel
112           else
113             # angle from rx to int in radians
114             gamma = acos(xi/sqrt(xi**2 + di**2))
115             if gamma >= alpha
116               los = false
```

```
117                 break
118               end
119             end
120           }
121         end
122         ret[i] = los
123       }
124       return ret
125     end
126
127     # Hata-based prop models assume that:
128     # (a) the tx is higher than the rx
129     # (b) rx is in [1,10]
130     # (c) tx is in [30,10]
131     #
132     # Here, we subtract off the minimum so
133     # that the heights are relative.
134     #
135     # Then, we swap them if the rx was higher (since
136     # loss is proportional, we'll just pretend
137     # we're transmitting from the rx to tx and
138     # calculate the loss, or something).
139     #
140     # We hard-code the rx to 1.0 (and adjust tx as such)
141     #
142     # Finally, we crudely lower or raise the resulting
143     # tx to make sure it's in the right range.
144     def hata_fix_heights(h1,h2)
145       m = [h1,h2].min
146       h1 -= m
147       h2 -= m
148       h2,h1 = [h1,h2].sort
149       h2 = 1.0
150       h1 -= 1.0
151       h1 = [h1,200.0].min
152       h1 = [h1,30.0].max
153       return h1,h2
154     end
155
156     ##### Path Loss Model Functions - All of these return an array of Path Loss
157     ##### Components (up to 4 of them) which are assumed to be attenuation in dB
158     ##### (i.e. positive means it is loss and negative means it is gain. Aside
159     ##### from model specific parameters, they should all "work" the same. They
160     ##### also all log warning messages to @warn and it's expected you use the
161     ##### warnage function (above) to clear out this array after each is used.
162
163     ############### BASIC MODELS ########################
164
165     # Simplified Egli Model
166     #
167     # Egli, John J. (Oct. 1957). "Radio Propagation above 40 MC over Irregular
168     # Terrain". Proceedings of the IRE (IEEE) 45 (10): 1383  1391. ISSN 0096-8390.
169     #
170     # Simplified version due to:
171     #
172     # Deslisle G.Y., Lefevre J., Lecours M., and Chouinard, J. Propagation loss
173     # prediction: a comparative study with application to the mobile radio channel.
174     # IEEE Trans. Veh. Tech. 1985. 26. 4. p. 295-308.
175     #
176     # This version presented in:
177     #
178     # Les Barclay. Propagation of Radiowaves. IEE. 2003. p. 209
179     #
180     # f is in MHz
181     def egli(f)
182       hb = self.tx.h # m
183       hm = self.rx.h # m
```

```
184       @warn.push "Frequency #{f} is out of the Egli Model's coverage" if f > 3000
185         or f < 30
186       @warn.push "Mobile receiver height lies at model discontinuity" if hm == 10
187       d = distance(self.tx,self.rx) # Km
188       lm = 76.3 - ((hm < 10) ? 10.0*log10(hm) : 20.0*log10(hm))
189       l = 40.0*log10(d) + 20.0*log10(f) - 20.0*log10(hb)
190       lfs = freespace(f).sum
191       [(l < lfs) ? lfs : l] # use freespace if our prediction is less than it
192     end
193
194     # Walfish-Ikegami model
195     #
196     # Ikegami proposed calculating the diffraction over each building in a path.
197     # The Walfish-Ikegami model assumes a regular grid of rectangular buildings,
198     # but otherwise makes the same computations.
199     #
200     # From: Les Barclay. Propagation of Radiowaves. IEE. 2003. p. 197
201     #
202     # (and numerous others)
203     #
204     # f is in MHz
205     # los is boolean (line-of-sight)
206     # h1 is in m
207     # h2 is in m
208     # hb is the nominal height of building roofs in m
209     # b is the nominal building separation in m
210     # w is the nominal street width in m
211     # phi is the angle of incident wave with respect to street in degrees
212     # city_size can be :medium or :large
213     #
214     # Default parameters provided on p. 152 of Barclay
215     #
216     # For 800 to 2000 MHz
217     def walfish(f,los,hb,b=20.0,w=10.0,phi=90,city_size=:medium)
218       h1 = self.tx.h
219       h2 = self.rx.h
220       d = distance(self.tx,self.rx)
221       model_name = "Walfish-Ikegami"
222       @warn.push "#{f} MHz is outside the #{model_name} model's coverage"
223         if f > 2000.0 or f < 800.0
224       @warn.push "#{d} Km is further than the #{model_name} model can support"
225         if d > 5 or d < 0.02
226       if los
227         return [freespace(f).sum,6*log10(d*50)]
228       else
229         dhb = h1 - b
230         dhm = hb - h2
231
232         # First, calculate the Roof-to-Street diffraction and scatter loss, rts
233         # ori is the orientation loss
234         ori = nil
235         if phi >= 0.0 and phi <= 35.0
236           ori = -10.0 + 0.354*phi
237         elsif phi >= 25.0 and phi <= 55.0
238           ori = 2.5 + 0.075*(phi-35)
239         elsif phi >= 55.0 and phi <= 90.0
240           ori = 4.0 - 0.144*(phi-55)
241         end
242         rts = -16.9 - 10*log10(w) + 10*log10(f) + 20*log10(dhm.abs) + ori
243
244         # Then, calculate the Multiscreen Diffraction loss, msd
245         # bsh is the shadowing gain that occurs when the base-station is higher
246         # than the rooftops in the msd calculation
247         bsh = (dhb <= 0) ? 0 : -18.0*log10(1 + dhb.abs)
248         ka = 54.0
249         if dhb <= 0 and d >= 0.5
250           ka += 0.8*dhb.abs
```

```
251         elsif dhb <= 0 and d < 0.5
252           ka += 0.8*dhb.abs*(d/0.5)
253         end
254         # kd is the distance factor in msd calculation
255         kd = 18.0
256         kd += 17*(dhb.abs/h1) if dhb <= 0.0
257         # kf is the frequency factor in msd calculation
258         kf = -4.0
259         kf += (city_size == :large) ? 1.5*(f/925.0 - 1) : 0.7*(f/925.0 - 1)
260         msd = bsh + ka + kd*log10(d) + kf*log10(f) - 9*log10(b)
261
262         # Finally, return the calculated path loss if it seems legal
263         if rts + msd >= 0.0
264           return [freespace(f).sum,rts,msd]
265         else
266           return [freespace(f).sum]
267         end
268       end
269     end
270
271     # COST-Hata/COST-231 model/Extended Hata Model
272     #
273     # http://www.iucaf.org/sschool/procs/propag.pdf
274     # http://en.wikipedia.org/wiki/COST_Hata_model
275     #
276     # f is in MHz
277     # city_size can be :medium, :large
278     # d is in km
279     # h1 & h2 are in m
280     #
281     # For 1500 to 2000 MHz
282     def cost_hata(f,city_size=:medium)
283       h1 = self.tx.h
284       h2 = self.rx.h
285       d = distance(self.tx,self.rx)
286
287       model_name = "Cost-231/Cost-Hata"
288       @warn.push "#{f} MHz is outside the #{model_name} model's coverage"
289         if f > 2000.0 or f < 1500.0
290       @warn.push "#{d} Km is further than the #{model_name} model can support"
291         if d > 20 or d < 1
292       @warn.push "#{h1} m (tx-height) is too high or low for the model"
293         if h1 < 30 or h1 > 200
294       @warn.push "#{h2} m (tx-height) is too high or low for the model"
295         if h2 < 1 or h2 > 10
296
297       h1,h2 = hata_fix_heights(h1,h2)
298       # crudely "round" down or up the freq
299       f = [f,1500.0].max
300       f = [f,2000.0].min
301
302       a = (1.1*log10(f) - 0.7)*h2 - (1.56*log10(f) - 0.8)
303       c = (city_size == :large) ? 3.0 : 0.0
304       [46.33 + 33.9*log10(f) - 13.82*log10(h2) - a + (44.9
305         - 6.55*log10(h2))*log10(d) + c]
306     end
307
308     # Hata-Okumura Model
309     #
310     # http://w3.antd.nist.gov/wctg/manet/calcmodels_dstlr.pdf
311     # http://w3.antd.nist.gov/cgi-bin/req_propcalc_tar.pl
312     # http://en.wikipedia.org/wiki/Hata_Model_for_Urban_Areas
313     #
314     # f is in Mhz
315     # h1 & h2 are in m
316     # d is in km
317     # city size can be :open, :suburban, :medium, :large
```

```
318     #
319     # For 150-1500 MHz
320     def hata(f,city_size=:medium,suppress_warnings=false,dont_fix_heights=false)
321       h1 = self.tx.h
322       h2 = self.rx.h
323       d = distance(self.tx,self.rx)
324       unless suppress_warnings
325         model_name = "Hata-Okumura"
326         @warn.push "#{f} MHz is outside the #{model_name} model's coverage"
327           if f > 1500.0 or f < 150.0
328         @warn.push "#{d} Km is further than the #{model_name} model can support"
329           if d > 10 or d < 1
330         @warn.push "#{h1} m (tx-height) is too high or low for the model"
331           if h1 < 30 or h1 > 200
332         @warn.push "#{h2} m (tx-height) is too high or low for the model"
333           if h2 < 1 or h2 > 20
334       end
335
336       h1,h2 = hata_fix_heights(h1,h2) unless dont_fix_heights
337       # crudely "round" down or up the freq
338       f = [f,150.0].max
339       f = [f,2000.0].min
340
341       a = (city_size == :large) ? 3.2*log10(pow(11.75*h2,2.0)) - 4.97 :
342         (1.1*log10(f) - 0.7)*h2 - (1.56*log10(f) - 0.8)
343       k = 0.0
344       if city_size == :suburban
345         k = 2*pow(log10(f/28.0),2.0) + 5.4
346       elsif city_size == :open
347         k = 4.78*pow(log10(f),2.0) - 18.33*log10(f) + 40.94
348       end
349       [69.55 + 26.16*log10(f) - 13.82*log10(h1) - a +
350         (44.9-6.55*log10(h1))*log10(d) - k]
351     end
352
353     # ITU-R/CCIR Model
354     #
355     # building percent should be in [0,100]
356     # f is in Mhz
357     # h1 & h2 are in m
358     # d is in km
359     def itu_r(f,building_percent=20.0)
360       h1 = self.tx.h
361       h2 = self.rx.h
362       d = distance(self.tx,self.rx)
363       model_name = "ITU-R/CCIR"
364       @warn.push "#{f} MHz is outside the #{model_name} model's coverage"
365         if f > 2000.0 or f < 1500.0
366       @warn.push "#{d} Km is further than the #{model_name} model can support"
367         if d > 10 or d < 1
368       @warn.push "#{h1} m (tx-height) is too high or low for the model"
369         if h1 < 30 or h1 > 200
370       @warn.push "#{h2} m (tx-height) is too high or low for the model"
371         if h2 < 1 or h2 > 10
372
373       h1,h2 = hata_fix_heights(h1,h2)
374       # crudely "round" down or up the freq
375       f = [f,1500.0].max
376       f = [f,2000.0].min
377
378       a = (1.1*log10(f) - 0.7)*h2 - (1.56*log10(f) - 0.8)
379       b = (building_percent == 0.0) ? 0.0 : 30 - 25*log10(building_percent)
380       [69.55 + 26.16*log10(f) - 13.82*log10(h1) - a +
381         (44.9 - 6.55*log10(h1))*log10(d) - b]
382     end
383
384     # Hata-Davidson Model
```

```
385     #
386     # http://w3.antd.nist.gov/wctg/manet/calcmodels_r1.pdf
387     #
388     # f is in MHz
389     # city_size is same as for hata
390     # this is just hata with some corrections for long links mostly
391     def hata_davidson(f, city_size=:medium)
392       h1 = self.tx.h
393       h2 = self.rx.h
394       d = distance(self.tx, self.rx) # in km
395
396       model_name = "Hata-Davidson"
397       @warn.push "#{d} Km is further than the #{model_name} model can support"
398         if d > 300 or d < 1
399       @warn.push "#{h1} m (tx-height) is too high or low for the model"
400         if h1 < 30.0 || h1 > 2500.0
401       @warn.push "#{f} MHz is outside the #{model_name} model's coverage"
402         if f > 1500.0 or f < 150.0
403       @warn.push "#{h2} m (rx-height) is too high or low for the model"
404         if h2 < 1 or h2 > 20
405
406       a = (d >= 20) ? 0.62137*(d - 20.0)*(0.5 + 0.15*log10(h1/121.92)) : 0.0
407       s1 = (d >= 64.38) ? 0.174*(d - 64.38) : 0.0
408       s2 = (h1 > 300) ? 0.00784*log10(9.98/d).abs*(h1 - 300.0) : 0.0
409       s3 = (f/250.0)*log10(1500.0/f)
410       s4 = (d > 64.38) ? (0.112*log10(1500.0/f)*(d - 64.38)) : 0.0
411
412       hata(f, city_size, true, false) + [a - s1 - s2 - s3 - s4]
413     end
414
415     # Green-Obaidat model.
416     #
417     # From: "An Accurate Line of Sight Propagation Performance Model for Ad-hoc
418     # 802.11 Wireless (WLAN) Devices". 2002.
419     #
420     # Is basically freespace pathloss with a correction for antenna heights...
421     #
422     # f is in MHz
423     # d is in Km
424     # h1 & h2 are in m
425     def green_obaidat(f)
426       h1 = self.tx.h
427       h2 = self.rx.h
428       d = distance(self.tx, self.rx)
429       [40.0*log10(d),20.0*log10(f),-20.0*log10(h1*h2)]
430     end
431
432     # Flat Edge Model
433     #
434     # From: S.R. Saunders and F.R. Bonar. Explicit Multiple building diffraction
435     # attenuation function for mobile radio wave propagation. Electron. Lett. 1991.
436     # 27 (14). p. 1276-1277.
437     #
438     # This version, with some approximations from:
439     # Les Barclay. Propagation of Radiowaves. IEE. 2003. p. 197
440     #
441     # n is number of buildings between tx and rx
442     # h0 is nominal height of a building
443     # w is the distance between buildings (or maybe the width of a building,
444     # it doesn't really matter)
445     #
446     # All distances are in meters unless otherwise specified
447     def flat_edge(f,n=5,h0=20,w=10)
448       lr = 0.25 # the refraction loss
449       hm = self.rx.h
450       @warn.push "Receiver height (#{hm}) is above assumed building height (#{h0})"
451         if hm > h0
```

```
452       # angle between ground and tx->rx LOS path
453       phi = deg_to_rad(ascension(self.rx,self.tx).abs)
454       lf = freespace(f).sum
455       wl = freq_to_wavelength(f/1000.0)
456       c1 = 3.29
457       c2 = 9.90
458       c3 = 0.77
459       c4 = 0.26
460       t = (phi*sqrt((PI*w)/wl)).abs
461       ln = 0
462       if t < 0 and t >= -1 and n >= 1 and n <= 100
463         # This is an approximate fit due to Barclay which he claims is
464         # accurate to less than +/- 1.5dB for 1<=n<=100 and -1<=t<0.
465         ln = -(c1 + c2*log10(t) - (c3 + c4*log10(n)))
466       else
467         # These fresnel approximations due to Saunders. Antennas and Propagation
468         # for Wireless Communication Systems. Appendix B.3.
469         fres_fu = Proc.new{ |u| (1.0 + 0.926*u)/(2.0 + 1.792*u + 3.104*(u**2)) }
470         fres_gu = Proc.new{ |u| 1.0/(2.0 + 4.142*u + 3.492*(u**2) + 6.670*(u**3)) }
471         fres_cu = Proc.new{ |u| 0.5 + fres_fu.call(u)*sin((PI/2)*(u**2)) -
472           fres_gu.call(u)*cos((PI/2)*(u**2)) }
473         fres_su = Proc.new{ |u| 0.5 - fres_fu.call(u)*cos((PI/2)*(u**2)) -
474           fres_gu.call(u)*sin((PI/2)*(u**2)) }
475         fs = Proc.new{ |jx| (exp(Complex(0,-jx**2))/(sqrt(Complex(0,2))))*
476                             ((fres_su.call(jx.real*sqrt(2.0/PI)) + 0.5) +
477                             Complex(0,fres_cu.call(jx.real*sqrt(2.0/PI)) + 0.5)) }
478         lnt = Proc.new{ |n,t| n == 0 ? 1.0 : (1.0/n)*(0..n-1).inject(0.0){ |sum,m|
479           sum + lnt.call(m,t)*fs.call(Complex(0,-t)*sqrt(n-m)) } }
480         ln = lnt.call(n,t).abs
481       end
482       # This equation is a simple knife-edge diffraction loss due to the Ikegami model
483       le = 0
484       le = 10.0*log10(f) + 10.0*log10(sin(deg_to_rad(phi))) + 20.0*log10(h0-hm) -
485         10.0*log10(w) - 10.0*log10(1.0 + 3.0/(lr**2)) - 5.8 if hm < h0 and
486           phi != 0
487       [ln,lf,le]
488     end
489
490     # Walfisch-Bertoni Model
491     #
492     # Much like the Flat Edge model, but assumes "many" buildings.
493     #
494     # From: Walfisch J. and Bertoni H.L.. A theoretical model of UHF propagation
495     # in urban environments. IEEE Trans. Ant. Prop. 1988. 36. (12) p. 1788-1796
496     #
497     # h0 is nominal height of a building (m)
498     # w is the distance between buildings (m)
499     def walfish_bertoni(f,h0=20,w=10)
500       d = distance(self.tx,self.rx)
501       hb = self.tx.h
502       hm = self.rx.h
503       @warn.push "Receiver height (#{hm}) is above assumed building height (#{h0})"
504         if hm > h0
505       la = (hm > h0) ? 0.0 : 5.0*log10((w/2) + (h0-hm)**2) - 9.0*log10(w) +
506         20.0*log10(atan((2.0*(h0-hm))/w))
507       lex = 57.1 + log10(f) + 18.0*log10(d) - ((hb-h0 > 0) ? 18.0*log10(hb-h0) :
508         0.0) - 18.0*log10(1.0 - d**2/(17.0*(hb-h0)))
509       [freespace(f),lex,la]
510     end
511
512     # Riback-Medbo Model
513     #
514     # From: M. Riback, J. Medbo, J.E. Berg, F. Harrysson, and H. Asplund. Carrier
515     # Frequency Effects on Path Loss. 20006.
516     #
517     # Attempts to provide a correction for using a given model from one frequency
518     # domain to predict PL values at a different frequency.
```

```
519     #
520     # f is the frequency WE are modeling in MHz
521     # f0 is the frequency the model we want to use was based on in MHz
522     # lf0 is the PL predicted by this model at the f0 frequency
523     def riback_medbo(f,f0,lf0)
524       # fitted constants
525       a = 0.09
526       b = 256*(10**6)
527       c = 1.8
528       k = a*(atan(f0/b - c) - atan(f/b - c)) # correction factor
529       return [lf0,20.0*log10(f/f0),-k*(lf0 - freespace(f0,2.0).sum)]
530     end
531
532     # Building-Transmission Model
533     #
534     # From: Y.L.C. de Jong, M. H. J. L. Koelen, and M. H. A. J. Herben. A
535     # Building-Transmission Model for Improved Propagation Prediction
536     # in Urban Microcells. IEEE Transactions on Vehicular Technology. Vol 53.
537     # No. 2. March, 2004.
538     #
539     # Predicts average loss due to transmitting "through" buildings
540     # This is for 1.9 GHz and must be used in combination with some other path-loss
541     # or ray-tracing model
542     def building_transmission(config_filename="config.yaml")
543       datasource = "buildings"
544       conductivity = 0.0
545       permitivity = 5.0
546       alpha = 2.1 # average attenuation (in dB) per meter inside building
547       sum = 0.0
548       intersections(config_filename,datasource){ |din|
549         # assuming 90-degree angles of incidence
550         theta0 = Math::PI/2.0 # angle relative to building surface going in
551         theta1 = Math::PI/2.0 # angle relative to building surface going out
552         # eq. 5
553         r0 = (sin(theta0)-sqrt(permitivity-cos(theta0)**2)) /
554           (sin(theta0)+sqrt(permitivity-cos(theta0)**2))
555         r1 = (sin(theta1)-sqrt(permitivity-cos(theta1)**2)) /
556           (sin(theta1)+sqrt(permitivity-cos(theta1)**2))
557         # eq. 7
558         t0 = sqrt(1 - r0.abs**2)
559         t1 = sqrt(1 - r1.abs**2)
560         # eq. 10
561         sum += alpha*din - 20.0*log10(t0) -  20.0*log10(t1)
562       }
563       return [sum]
564     end
565
566     # Gas Attenuation Model
567     #
568     # Computes additional attenuation due to transmission through water vapor
569     # within oxygen. Note that this is for sea-level and that the ITU
570     # recommendation is to not bother for f < 10 GHz.
571     #
572     # For the sort of distances and frequencies we're working with, this is an
573     # attenuation of like 0.01 dB. Not really worth considering...
574     #
575     # From ITU-R P.676.
576     #
577     # ITU-R P.836 gives information on water vapor density.
578     #
579     # ITU-R P.452-13 gives some description fo this too
580     #
581     # http://www.mike-willis.com/Tutorial/PF5.htm
582     #
583     # p is water vapour concentration in grams per cubic meter
584     # P.452 suggests that you can use p = 7.5 + 2.5*omega
585     # where omega is the fraction of the total path over water
```

```
586     def gas_attenuation(f,p=7.5)
587       d = distance(self.tx,self.rx) # in Km
588       f = f/1000.0 # in GHz
589       a_w = (0.0050 + 0.0021*p + 3.6/((f - 22.2)**2  + 8.5) + 10.6/
590         ((f - 183.3)**2 + 9.0) + 8.9/((f - 325.4)**2 + 26.3))*(f**2)*p*0.001
591       a_o2 = (7.19*0.01 + 6.09/(f**2 + 0.277) + 4.81/
592         ((f - 57.0)**2 + 1.5))*(f**2)*0.01
593       [a_w*d,a_o2*d]
594     end
595
596     # Standford University Interim Model (SUI)
597     #
598     # Note that this is a less-complex precurser to the Erceg-Greenstrein Model
599     #
600     # From:
601     #
602     # Abhayawardhana \textit{et al.} Comparison of Empirircal Propagation Path
603     # Loss Models for Fixed Wireless Access Systems.
604     #
605     # and
606     #
607     # Erceg \textit{et al.} Channel Models for Fixed Wireless Applications. Tech.
608     # Report. IEEE 802.16 Broadband Wireless Access Working Group. January 2001.
609     #
610     # f is in MHz
611     # terrain type can be :a, :b, or :c
612     #   from the paper: The maximum path loss category is hilly terrain with
613     #   moderate-to-heavy tree densities (Category A). The minimum path
614     #   loss category is mostly flat terrain with light tree densities
615     #   (Category C). Intermediate path loss condition is captured in
616     #   Category B.
617     def sui(f,terrain_type=:a,vary=false)
618       hb = self.tx.h
619       hr = self.rx.h
620       d0 = 100.0 # m
621       d = distance(self.tx,self.rx)*1000.0 # m
622       wl = freq_to_wavelength(f/1000.0)
623       biga = 20.0*log10((4.0*PI*d0)/wl)
624       a = {:a => 4.65, :b => 4.0, :c => 3.6}[terrain_type]
625       b = {:a => 0.0075, :b => 0.0065, :c => 0.005}[terrain_type]
626       c = {:a => 12.6, :b => 17.1, :c => 20.0}[terrain_type]
627       xf = 6.0*log10(f/2000.0)
628       gamma = a - b*hb + c/hb
629       xh = (terrain_type == :c) ? -20.0*log10(hr/2000.0) : -10.8*log10(hr/2000.0)
630       s = vary ? rlognorm(0.0,runif(8.2,10.6)) : 0.0
631       [biga,10.0*gamma*log10(d/d0),xf,xh,s]
632     end
633
634     # ECC-33 Model
635     #
636     # From:
637     #
638     # Abhayawardhana \textit{et al.} Comparison of Empirircal Propagation Path
639     # Loss Models for Fixed Wireless Access Systems.
640     #
641     # f is in MHz
642     # city_size can be large or medium
643     def ecc33(f,city_size=:large)
644       f = f/1000.0 # GHz
645       hb = self.tx.h
646       hr = self.rx.h
647       d = distance(self.tx,self.rx) # km
648       afs = 92.4 + 20.0*log10(d) + 20.0*log10(f)
649       abm = 20.41 + 9.83*log10(d) + 7.894*log10(f) + 9.56*(log10(f))**2
650       gb = log10(hb/200.0)*(13.958 + 5.8*(log10(d))**2)
651       gr = (city_size == :medium) ? 0.0 : (42.57 + 13.7*log10(f))*(log10(hr) - 0.585)
652       [afs,abm,-gb,-gr]
```

```
653     end
654
655     # Edwards-Durkin Model
656     #
657     # From:
658     #
659     # G. Y. Delisle, J. P. Lefevre, M. Lecours, and J.Y. Choinard. Propagation
660     # Loss Prediction: A Comparative Study with Application to the Mobile Radio
661     # Channel. IEEE Trans on Vehicular Technology. Vol. VT-34. No. 2. May, 1985.
662     #
663     # f is carrier in MHz
664     # use_terrain decided whether we should compute diffraction over the path
665     def edwards_durkin(f,use_terrain=false,delta_h=15.0)
666       r = distance(self.tx,self.rx) # km
667       hb = self.tx.h
668       hm = self.rx.h
669       k1 = 32.45 # for isotropic ant.; use 28.85 for half-wave dipoles
670       k2 = 118.7 # for isotropic ant.; use 115.1 for half-wave dipoles
671
672       # lf is a lower bound which we won't use here because Delisle says lp + ld is a better
673       # fit to data in practice
674       #lf = k1 + 20.0*log10(f) + 20.0*log10(r) # classical freespace loss
675
676       lp = k2 - 20.0*log10(hm) - 20.0*log10(hb) + 40.0*log10(r) # plane earth loss
677       ld = use_terrain ? terrain_diffraction_estimate(f,delta_h).sum : 0.0
678
679       return [lp,ld]
680     end
681
682     # Blomquist-Ladell
683     #
684     # From:
685     #
686     # G. Y. Delisle, J. P. Lefevre, M. Lecours, and J.Y. Choinard. Propagation
687     # Loss Prediction: A Comparative Study with Application to the Mobile Radio
688     # Channel. IEEE Trans on Vehicular Technology. Vol. VT-34. No. 2. May, 1985.
689     #
690     # f is carrier in MHz
691     # use_terrain decided whether we should compute diffraction over the path
692     def blomquist_ladell(f,use_terrain=false,delta_h=15.0)
693       r = distance(self.rx,self.tx) # km
694       d = 1000.0*r # m
695       hb = self.tx.h
696       hm = self.rx.h
697       eb = em = 10.0 # permitivity
698       wl = freq_to_wavelength(f/1000.0)
699       lf = 32.45 + 20.0*log10(f) + 20.0*log10(r)
700       k = 4.0/3.0 # earth radius factor
701       a = 6.371*(10**6) # earth radius in m
702       x = ((2.0*PI/wl)**(1.0/3.0))*((k*a)**(-2.0/3.0))*d
703       y = (x < 0.53) ? -2.8*x : 6.7 + 10.0*log10(x) - 10.2*x
704       fb = 10.0*log10( (((4.0*PI*hb**2)/(wl*d)) + ((wl*(eb**2))/(PI*d*(eb-1)))) *
705                        (((4.0*PI*hm**2)/(wl*d)) + ((wl*(em**2))/(PI*d*(em-1)))) ) + y
706       ld = use_terrain ? terrain_diffraction_estimate(f,delta_h).sum : 0.0
707       lt = 0.0
708       if fb <= 0
709         [lf,sqrt(fb**2 + ld**2)]
710       elsif fb > 0 and fb <= ld.abs
711         [lf,sqrt(fb**2 - ld**2)]
712       elsif fb > 0 and fb > ld.abs
713         [lf,-sqrt(fb**2 - ld**2)]
714       end
715     end
716
717     # Alsebrook Parsons Model
718     #
719     # From:
```

```
720     #
721     # G. Y. Delisle, J. P. Lefevre, M. Lecours, and J.Y. Choinard. Propagation
722     # Loss Prediction: A Comparative Study with Application to the Mobile Radio
723     # Channel. IEEE Trans on Vehicular Technology. Vol. VT-34. No. 2. May, 1985.
724     #
725     # f is carrier in MHz
726     # use_terrain decided whether we should compute diffraction over the path
727     # delta_h is a terrain roughness parameter passed to
728     #   terrain_diffraction_estimate() if required
729     # h0 is the average height of buildings in m
730     # d2 is the average width of streets in m
731     def allsebrook_parsons(f,use_terrain=false,delta_h=15.0,h0=5.0,d2=20.0)
732       r = distance(self.rx,self.tx) # km
733       d = 1000.0*r # m
734       hb = self.tx.h
735       hm = self.rx.h
736       eb = em = 10.0 # permitivity
737       wl = freq_to_wavelength(f/1000.0)
738       lf = 32.45 + 20.0*log10(f) + 20.0*log10(r)
739
740       k = 4.0/3.0 # earth radius factor
741       a = 6.371*(10**6) # earth radius in m
742       x = ((2.0*PI/wl)**(1.0/3.0))*((k*a)**(-2.0/3.0))*d
743       y = (x < 0.53) ? -2.8*x : 6.7 + 10.0*log10(x) - 10.2*x
744       fb = 10.0*log10( (((4.0*PI*hb**2)/(wl*d)) + ((wl*(eb**2))/(PI*d*(eb-1)))) *
745                        (((4.0*PI*hm**2)/(wl*d)) + ((wl*(em**2))/(PI*d*(em-1)))) ) + y
746
747       ld = use_terrain ? terrain_diffraction_estimate(f,delta_h).sum : 0.0
748
749       gamma = (f > 200) ? 13.0 : 0.0
750       lb = (h0 > hm) ? 20.0*log10((h0-hm)/(548.0*sqrt(d2*0.01*f))) + 16.0 : 0.0
751
752       [lf,sqrt(fb**2+ld**2),lb,gamma]
753     end
754
755     # Rural Hata/Medeisis-Hata model
756     #
757     # From:
758     #
759     # A. Medeisis and A. Kajackas. On the Use of the Universal Okumura-Hata
760     # Propagation Prediction Model in Rural Areas. IEEE Vehicular Technology
761     # Conference Proceedings. 2000-Spring. Tokyo. 1815-1818.
762     #
763     # env can be :rural or :urban
764     #
765     def rural_hata(f,config,env=:rural,dont_fix_heights=false)
766       hms = self.rx.h # m
767       hbs = self.tx.h # m
768       hbs,hms = hata_fix_heights(hbs,hms) unless dont_fix_heights
769       r = distance(self.tx,self.rx) # km
770       closest_f = nil
771       [160,450,900].each{ |f2|
772         closest_f = f2 if closest_f.nil? or (f-f2).abs < (closest_f-f).abs
773       }
774       e0 = { :rural => { 160 => 40.0, 450 => 40.0, 900 => 35.0 },
775              :urban => { 160 => 40.0, 450 => 50.0, 900 => 60.0 } }
776       gamma = { :rural => { 160 => 1.25, 450 => 1.30, 900 => 1.00 },
777                 :urban => { 160 => 1.20, 450 => 1.20, 900 => 1.25 } }
778
779       my_e0 = e0[env][closest_f]
780       my_gamma = gamma[env][closest_f]
781
782       # it's not clear if we should be using f or closest_f in these calcs. I think f is
783       # more better even though it conflicts with the f we use to select fitted params.
784
785
786       a = Proc.new { |hms,f| (1.1*log(f) - 0.7)*hms - (1.56*log(f) - 0.8) }
```

```
787       esys = -6.16*log(f) + 13.82*log(hbs) + a.call(hms,f)
788       gamma_sys = -my_gamma*(44.9 - 6.55*log(hbs))
789       loss_dBuVm = my_e0 + esys + gamma_sys*log(r)
790       # uhh...dBuV/m...
791       # http://www.microvolt.com/table.html
792       # http://www.softwright.com/faq/engineering/FIELD%20INTENSITY%20UNITS.html
793       #
794       g_rx = ant_gain(self.rx,self.tx,config)
795       loss_dBm = loss_dBuVm + g_rx - 20.0*log(f) - 77.0
796       [-loss_dBm]
797     end
798
799     # Oda Model
800     #
801     # From:
802     #
803     # Yasuhiro Oda and Koichi Tsunekawa. Advanced LOS Path Loss Model in Microwave
804     # Mobile Communications. 10th International Conference on Antennas and
805     # Propagation. 1997.
806     #
807     # A very minor correction to 2-ray path loss. Pretty dumb.
808     #
809     # h0 is average the height of street-level scatters such as traffic and signs
810     #   and mailboxes and whatnot
811     # s is the probability of collision per unit of distance (err...)
812     def oda(f,h0=1.0,s=0.5)
813       hb = self.tx.h
814       hm = self.rx.h
815       r = distance(self.rx,self.tx) # m
816       rrm = sqrt(r**2 + ((hb-h0)+(hm-h0))**2) # distance along relfected path in m
817       rt = sqrt(r**2 + (hb-hm)**2) # distance along LOS path in m
818       pr = exp(-s*r)
819       wl = freq_to_wavelength(f/1000.0) # wavelength
820       k = (2.0*PI)/wl # wave number
821
822       # This is from the paper, but it doesn't seem to work...
823       #a = Complex.new(0.0,-k*rt)
824       #b = Complex.new(0.0,-k*rrm)
825       #bigr = -1.0 # assumed reflection coefficient is exactly out of phase
826       #[10.0*log(pr*(wl/(4.0*PI))*((exp(a)/rt) + bigr*(exp(b)/rrm)).abs)]
827
828       # This is based on Saunders2007 and assumes R = -1.
829       # The paper gives no guidance, so this is probably fine.
830       hb -= h0
831       hm -= h0
832       [10.0*log(pr*2.0*((wl/(4.0*PI*r))**2)*(1.0 - cos((k*2.0*hm*hb)/r)))]
833     end
834
835     # deSouza-Lins
836     #
837     # From:
838     #
839     # R. S. deSouza and R. D. Lins. A New Propagation Model for 2.4 GHz Wireless LAN.
840     # APCC 2008.
841     #
842     # An explicitly data-fitted model which includes relative humidity. Probably
843     # doesn't work for anything longer than about 120m since that is as far away
844     # as they got from the AP in measurement.
845     def desouza_lins(f,rh=50.0)
846       d = distance(self.tx,self.rx)*1000.0 # m
847       b0 = [38.88,37.67].mean
848       b1 = [25.849,15.402].mean
849       b2 = [0.099,0.155].mean
850       b3 = [7.508,11.56].mean
851       [b0,b1*log(d),b2*d,b3*log(rh)]
852     end
853
```

```
854     ################ TERRAIN MODELS ##########################
855
856     # ITU Terrain Model
857     #
858     # From:
859     #
860     # J.S. Seybold. Introduction to RF Propagation. p. 144-145
861     #
862     # Propagation data and prediction methods required for the design of
863     # terrestrial line-of-sight systems. Recommendation ITU-R P.530-11.
864     #
865     def itu_terrain(f,path)
866       d = distance(self.tx,self.rx)*1000 # in m
867       wavelength = freq_to_wavelength(f/1000)
868       agl_rx = rx.ele+rx.h                  # in m
869       agl_tx = tx.ele+tx.h                  # in m
870       los_slope = (agl_tx-agl_rx)/d         # negative if tx is above rx
871       max_obstruction = -1.0
872       los_max_obstruction = nil
873       fres_max_obstruction = nil
874
875       (1..self.length-2).each{ |i|
876         int = self[i]
877         next if rx.ele.nil? or tx.ele.nil? or int.ele.nil?
878         di = distance(self.tx,int)*1000       # in m
879         agl_int = int.ele+int.h                  # in m
880         los_elev = di*los_slope + agl_tx
881         fresnel_radius = sqrt((wavelength*di*(d-di))/(di+(d-di)))
882         fresnel_lower = los_elev - fresnel_radius
883         if agl_int > fresnel_lower and agl_int > max_obstruction
884           max_obstruction = agl_int
885           los_max_obstruction = los_elev
886           fres_max_obstruction = fresnel_radius
887         end
888       }
889       a = 0.0
890       if max_obstruction >= 0
891         h = los_max_obstruction - max_obstruction
892         a = (-20.0*h)/fres_max_obstruction + 10.0
893         # negative and small losses are not realistic according to Seybold
894         a = 0.0 if a <= 6
895       end
896       return [freespace(f,2.0).sum,a]
897     end
898
899     # From: ITU-R P.452
900     #
901     # p is the percentile not-to-exceed, so 50 means this is the median value.
902     # 100 would be a worst-case and 0 a best-case.
903     #
904     # delta_n is radio refractivity of the earth. some values:
905     #   35 - boulder average
906     #   50 - hamilton average
907     #   40 - portland average, boulder worst
908     #   45 - portland worst
909     #   60 - hamilton worst
910     # n0 is the sea level surface refractivity. some values:
911     #   300 - boulder
912     #   320 - portland
913     #   340 - hamilton
914     # omega is the fraction of the total path over water
915     def itu_r_452(f,config,p=50.0,delta_n=40.0,n0=320.0,omega=0.0)
916       d = distance(self.tx,self.rx)   # in km
917       k50 = 157.0/(157.0 - delta_n)
918       kbeta = 3.0
919       ae = 6371.0*k50
920       abeta = 6371.0*kbeta
```

```
921       hts = self.tx.z # height above mean sea level (m)
922       hrs = self.rx.z # m
923       phi = self.center_latitude
924       lambda1 = freq_to_wavelength(f/1000)
925       ha = @path.collect{ |int| int.h + int.ele }.mean # mean height
926
927       ### PATH CLASSIFICATION CALC (Appendex 2. Sect. 4 and 5.1.1 - 5.1.5) ###
928
929       theta_td = (hrs-hts)/d - (1000*d)/(2.0*ae)
930       theta_max = 0.0
931       theta_t = 0.0      # transmitter antenna horizon elevation angle (mrad)
932       theta_r = 0.0      # receiver antenna horizon elevation angle (mrad)
933       dlt = 0.0
934       ilr = 0
935       dlr = 0.0
936       ilt = 0
937
938       (1..self.length-2).each{ |i| # loop over path omitting tx and rx
939         int = self[i] # intermediary point
940         hi = int.z # m
941         di = distance(self.tx,int) # km
942
943         theta_i = (hi-hts)/di - (1000*di)/(2.0*ae)
944         theta_j = (hi-hrs)/(d-di) - (1000*(d-di))/(2.0*ae)
945
946         if theta_max.nil? or theta_i > theta_max
947           theta_max = theta_i
948           theta_t = theta_i
949           dlt = di
950           ilt = i
951         end
952         if theta_r.nil? or theta_j > theta_r
953           theta_r = theta_j
954           dlr = d - di
955           ilr = i
956         end
957       }
958
959       theta = (1000*d)/ae + theta_t + theta_r     # angular distance in mrad
960
961       # true if this is a trans-horizon path (unlikely for short distances)
962       transhorizon = theta_max > theta_td
963
964       ### DIFFRACTION CALCULATIONS (Sect. 4.2) ###
965
966       # IMPORTANT ASSUMPTION:
967       # assume the entire path is over land...
968       # this doesn't mean small bodies of water.
969       # This means, like the ocean and stuff.
970       dtm = dlm = dct = dcr = tau = 0
971
972       mu1 = (10**(-dtm/(16-6.6*tau)) + (10**-(0.496+0.354*tau))**5)**0.2
973       mu1 = 1.0 if mu1 > 1.0
974       mu4 = (phi <= 70) ? 10**((-0.953+0.0176*phi.abs)*log10(mu1)) :
975         10**(0.3*log10(mu1))
976
977       # point of incidence of anomolous propagation (%) for the path center location
978       beta0 = (phi <= 70) ? (10**(-0.015*phi.abs + 1.67))*mu1*mu4 : 4.17*mu1*mu4
979
980       # interpolation factor for path angular distance
981       fj = 1.0 - 0.5*(1.0 + tanh(3.0*0.8*((theta-0.3)/0.3)))
982       # interpolation factor for great circle path distance
983       fk = 1.0 - 0.5*(1.0 + tanh(3.0*0.5*((d-20)/20)))
984
985       # water vapor density and gaseous attenuation
986       p = 7.5 + 2.5*omega
987       ag = gas_attenuation(f,p).sum
```

```
988
989       esp = 2.6*(1.0 - exp(-0.1*(dlt+dlr)))*log(p/50)
990       esbeta = 2.6*(1.0 - exp(-0.1*(dlt+dlr)))*log(beta0/50)
991
992       # approximate inverse cumulative normal distribution (Appendix 3)
993       inv_cum_norm = Proc.new{ |x|
994         c = [2.515516698,0.802853,0.010328]
995         d = [1.432788,0.189269,0.001308]
996         t = sqrt(-2.0*log(x))
997         xi = ((c[2]*t + c[1])*t + c[0])/(((d[2]*t + d[1])*t + d[0])*t + 1.0)
998         xi - t
999       }
1000
1001      # approximate knife-edge diffraction loss eq. 13
1002      ke_diff_loss = Proc.new{ |v|
1003        v < -0.78 ? 0.0 : 6.9 + 20.0*log10(sqrt((v-0.1)**2 + 1) + v - 0.1)
1004      }
1005
1006      fi = 0
1007      if p == 50.0
1008        fi = 0
1009      elsif p > 50.0 and p < beta0
1010        fi = inv_cum_norm.call(p/100)/inv_cum_norm.call(beta0/100)
1011      elsif beta0 >= p
1012        fi = 1
1013      end
1014
1015      # basic transmission loss due to free-space propagation and attenuation by
1016      # atmospheric gasses (Sect. 4.1)
1017      lbfsg = 92.5 + 20.0*log(f/1000) + 20.0*log(d) + ag
1018
1019      # correction for overall path slope
1020      xi_m = cos(atan(0.01*(hrs-hts)/d))
1021      # principle edge diffraction parameter
1022      vm50 = 0.0
1023      im50 = 0
1024      him50 = 0.0
1025      dim50 = 0.0
1026      (1..@path.length-2).each{ |i|
1027        int = @path[i]
1028        di = distance(int,self.tx)
1029        his = int.z # height above mean sea level
1030        hi = his + 1000*(di*(d-di))/(2.0*ae) - (hts*(d-di)+hrs*di)/d
1031        val = xi_m*hi*sqrt((0.02*d)/(lambda1*di*(d-di)))
1032        if vm50.nil? or val > vm50
1033          vm50 = val
1034          im50 = i
1035          him50 = his
1036          dim50 = di
1037        end
1038      }
1039      lm50 = ke_diff_loss.call(vm50)
1040
1041      ld50 = 0.0
1042      ldbeta = 0.0
1043      lt50 = 0.0
1044      lr50 = 0.0
1045      ltbeta = 0.0
1046      lrbeta = 0.0
1047
1048      # only calculate ld50 and ldbeta if lm50 is nonzero
1049      if lm50 != 0.0
1050        # only calculate lt50 if there is a transmitter-side secondary edge
1051        if im50 <= 1
1052          xi_t = cos(atan(0.01*(him50-hts)/dim50))
1053          vt50 = nil
1054          it50 = nil
```

```
1055          hit50 = nil
1056          dit50 = nil
1057          (1..im50-1).each{ |i|
1058            int = @path[i]
1059            di = distance(int,self.tx)
1060            his = int.z
1061            hi = his + 1000*(di*(dim50-di))/(2.0*ae) -
1062              (hts*(dim50-di)+him50*di)/dim50
1063            val = xi_t*hi*sqrt((0.02*dim50)/(lambda1*di*(dim50-di)))
1064            if vt50.nil? or val > vt50
1065              vt50 = val
1066              it50 = i
1067              hit50 = his
1068              dit50 = di
1069            end
1070          }
1071          lt50 = (im50 >= 2) ? ke_diff_loss.call(vt50) : 0.0
1072
1073          if lt50 != 0.0
1074            hitbeta = hit50 + 1000*(dit50*(dim50-dit50))/(2.0*abeta) -
1075              (hts*(dim50-dit50)+him50*dit50)/dim50
1076            vtbeta = xi_t*hitbeta*sqrt((0.02*dim50)/(lambda1*dit50*(dim50-dit50)))
1077            ltbeta = ke_diff_loss.call(vtbeta)
1078          end
1079        end
1080
1081        # only calculate lr50 if there is a receiver-side secondary egde
1082        if im50 < @path.length-2
1083          xi_r = cos(atan(0.01*(hrs-him50)/(d-dim50)))
1084          vr50 = nil
1085          ir50 = nil
1086          hir50 = nil
1087          dir50 = nil
1088          (im50+1..@path.length-2).each{ |i|
1089            int = @path[i]
1090            di = distance(int,self.tx)
1091            his = int.z
1092            hi = his + 1000*((di-dim50)*(d-di))/(2.0*ae) - (him50*(d-di) +
1093              hrs*(di-dim50))/(d-dim50)
1094            val = xi_r*hi*sqrt((0.02*(d-dim50))/(lambda1*(di-dim50)*(d-di)))
1095            if vr50.nil? or val > vr50
1096              vr50 = @path.inject(0.0){ |r,int|
1097                hi = int.h + int.ele
1098                dis = distance(int,self.tx)
1099                r += (hi-ha)*(dis-(d/2.0)) }
1100              ir50 = i
1101              hir50 = his
1102              dir50 = di
1103            end
1104          }
1105          lr50 = (im50 < @path.length-2) ? ke_diff_loss.call(vr50) : 0.0
1106          if lr50 != 0.0
1107            hirbeta = hir50+1000*((dir50-dim50)*(d-dir50))/(2.0*abeta) -
1108              (him50*(d-dir50)+hrs*(dir50-dim50))/(d-dim50)
1109            vrbeta = xi_r*hirbeta*sqrt((0.02*(d-dim50))/
1110              (lambda1*(dir50-dim50)*(d-dir50)))
1111            lrbeta = ke_diff_loss.call(vrbeta)
1112          end
1113        end
1114
1115        # finally calculate ld50 from lt50 and lr50 and lm50
1116        ld50 = lm50 + (1-exp(-lm50/6.0))*(lt50 + lr50 + 10.0 + 0.04*d)
1117
1118        # then the beta stuff...
1119        himbeta = him50 + 1000*(dim50*(d-dim50))/(2.0*ae) -
1120          (hts*(d-dim50)+hrs*dim50)/d
1121        vmbeta = xi_m*himbeta*sqrt((0.02*d)/(lambda1*dim50*(d-dim50)))
```

```
1122        lmbeta = ke_diff_loss.call(vmbeta)
1123
1124        ldbeta = lmbeta + (1.0 - exp(-lmbeta/6.0))*(ltbeta + lrbeta + 10.0 +0.04*d)
1125      end
1126
1127      ### TROPOSPHERIC SCATTER CALCULATIONS (Sect. 4.3)
1128
1129      gt = ant_gain(self.tx,self.rx,config)
1130      gr = ant_gain(self.rx,self.tx,config)
1131
1132      # aperture to medium coupling loss
1133      lc = 0.051*exp(0.055*(gt+gr))
1134      # frequency dependent loss
1135      lf = 25.0*log10(f) - 2.5*log10(f/2)**2
1136      # basic transmission loss due to troposcatter
1137      lbs = 190.0 + lf + 20.0*log10(d) + 0.573*theta - 0.15*n0 + lc + ag -
1138        10.1*(-log10(p/50.0))**0.7
1139
1140      ### DUCTING/LAYER-REFLECTION CALCULATIONS (Sect. 4.2)
1141
1142      theta_t1 = (theta_t <= dlt) ? theta_t : 0.1*dlt
1143      theta_r1 = (theta_r <= dlr) ? theta_r : 0.1*dlr
1144
1145      theta1 = (1000*d)/ae + theta_t1 + theta_r1
1146
1147      theta_t2 = theta_t - 0.1*dlt
1148      theta_r2 = theta_r - 0.1*dlr
1149
1150      # over sea surface duct coupling corrections
1151      act = 0.0
1152      if omega >= 0.75 and dct <= dlt and dct <= 5.0
1153        act = -3.0*exp(-0.25*dct**2)*(1.0 + tanh(0.07*(50.0-hts)))
1154      end
1155      acr = 0.0
1156      if omega >= 0.75 and dcr <= dlr and dcr <= 5.0
1157        acr = -3.0*exp(-0.25*dcr**2)*(1.0 + tanh(0.07*(50.0-hrs)))
1158      end
1159
1160      ### SMOOTH EARTH MODEL CALCULATIONS (Appendix 2. Sect.5.1.6) ###
1161
1162      # slope of the smooth-earg surface relative to sea level
1163      # IMPORTANT ASSUMPTION: assume sample points are equally spaced.
1164      # there are other ways of calculating m if they are not
1165      mnum = 0.0
1166      mdem = 0.0
1167      (0..@path.length-1).each{ |i|
1168        hi = @path[i].z
1169        di = distance(@path[i],self.tx)
1170        mnum += (hi-ha)*(di-d/2.0)
1171        #puts "ha = #{ha}, di = #{di}, d = #{d}"
1172        mdem += (di-d/2.0)**2
1173      }
1174      m = mnum/mdem
1175
1176      #puts "m = #{mnum}/#{mdem} = #{m}"
1177
1178      hst = ha - m*d/2.0
1179      hsr = hst + m*d
1180
1181      recalc_m = false
1182      if hst > self.tx.z
1183        hst = self.tx.z
1184        recalc_m = true
1185      end
1186      if hsr > self.rx.z
1187        hsr = self.rx.z
1188        recalc_m = true
```

```
1189        end
1190        m = (hsr-hst)/m if recalc_m
1191
1192        # terrain roughness parameter
1193        hm = 0.0
1194        (ilt..ilr).each{ |i|
1195          dis = distance(@path[i],self.tx)
1196          hi = @path[i].z
1197          val = hi - (hst + m*dis)
1198          #puts "val = #{hi} - (#{hst} + #{m}*#{dis}"
1199          hm = val if hm.nil? or val > hm
1200        } unless ilr < ilt
1201        #puts "hm = #{hm}"
1202
1203        epsilon = 3.5 # not used
1204        alpha = 0 # because tau is zero based on IMPORTANT ASSUMPTION above
1205        mu2 = 1.0 # because alpha is zero
1206        mu3 = (hm <= 10) ? 1.0 : exp(-4.6*(10**-5)*(hm-10)*(43+6*([d-dlt-dlr,40.0].min)))
1207        beta = beta0*mu2*mu3
1208        #puts "beta = #{beta0}*#{mu2}*#{mu3} = #{beta}"
1209        gamma = (1.076/((2.0058-log10(beta))**1.012))*
1210          exp(-(9.51-4.8*log10(beta)+0.198*(log(beta)**2))*(10**-6)*(d**1.13))
1211        ap = -12.0 + (1.2 + 0.037*d)*log10(p/beta) + 12.0*(p/beta)**gamma
1212        gamma_d = 0.0005*ae*(f**(1.0/3.0))
1213
1214        ad = gamma_d*theta1 + ap
1215
1216        # site shielding losses
1217        ast = 0.0
1218        asr = 0.0
1219        if theta_t2 > 0
1220          ast = 20.0*log10(1.0 + 0.361*theta_t2*sqrt(f*dlt)) +
1221            0.264*theta_t2*(f**(1.0/3.0))
1222        end
1223        if theta_r2 > 0
1224          asr = 20.0*log10(1.0 + 0.361*theta_r2*sqrt(f*dlr)) +
1225            0.264*theta_r2*(f**(1.0/3.0))
1226        end
1227
1228        # total fixed coupling losses (except for local clutter losses) between the
1229        # antennas and the anomolous propagation structure within the atmosphere
1230        af = 102.45 + 20.0*log10(f) + (dlt+dlr > 0.0 ? 20.0*log10(dlt+dlr) : 0.0) +
1231          ast + asr + act + acr
1232
1233        # basic transmission loss occuring during periods of anomalous propagation
1234        # (ducting and layer reflection)
1235        lba = af + ad + ag
1236
1237        ### ADDITIONAL CLUTTER LOSSES (Sect. 4.5)
1238
1239        # Note, it's note clear if these should be calculated over the total path
1240        # or just near the ends Also, if we calculate total path clutter for both
1241        # the receiver and transmitter, some double counting occurs. What I'm going
1242        # to do here is call aht the additional loss from the clutter on the tx side
1243        # of the path and ahr the addition loss from the receiver side of the path.
1244        # Each will be capped at 20dB as specified. If there's supposed to be a gap
1245        # in between, I'm not sure what it should be (i.e. how far away something
1246        # can be and still be considered "local clutter") maybe for microcell
1247        # networks, it's all relevant...
1248
1249        # Note also that this will count more clutter for more sample points,
1250        # which is maybe wrong. Really need to know what "percentage" of the path is
1251        # "local" clutter.  For now, we'll be conservative and count everything
1252
1253        aht = 0.0
1254        ahr = 0.0
1255        (0..@path.length-1).each{ |i|
```

```
1256          int = @path[i]
1257          next if int.clutter.nil?
1258          middle_i = (@path.length/2).floor
1259          tx_side = (i <= middle_i)
1260          h = tx_side ? self.tx.h : self.rx.h
1261          d = tx_side ? distance(self.tx,@path[i]) : distance(self.rx,@path[i])
1262          h_clutter,d_clutter = int.clutter
1263          val = 10.25*exp(-d_clutter)*(1.0-tanh(6.0*((h/h_clutter)-0.625))) - 0.33
1264          tx_side ? aht += val : ahr += val
1265        }
1266        aht = [aht,20.0].min
1267        ahr = [ahr,20.0].min
1268
1269        ### OVERALL PREDICTION (Sect. 4.6)
1270
1271        # diffraction loss not to exceed p%
1272        ldp = ld50 + fi*(ldbeta - ld50)
1273
1274        # median basic transmission loss associated with diffraction
1275        lbd50 = lbfsg + ld50
1276
1277        # basic transmission loss not to exceed for time
1278        # percentage p% due to LOS propagation
1279        lb0p = lbfsg + esp
1280
1281        # basic transmission loss not exceeded for the time percentage
1282        # beta0% due to LOS propagation
1283        lb0beta = lbfsg + esbeta
1284
1285        # basic transmission loss associated with diffraction not exceed p% of time
1286        lbd = lb0p + ldp
1287
1288        # notational minimum basic transmission loss for LOS propagation
1289        # and over-sea subpath diffraction
1290        lminb0p = (p < beta0) ? lb0p + (1-omega)*ldp : lbd50 +
1291          (lb0beta + (1-omega)*ldp - lbd50)*fi
1292
1293        # notational minimum basic transmission loss
1294        lminbap = 2.5*log(exp(lba/2.5) + exp(lb0p/2.5))
1295
1296        # notational basic transmission loss
1297        lbda = (lminbap > lbd) ? lbd : lminbap + (lbd-lminbap)*fk
1298
1299        # modified basic transmission loss
1300        lbam = lbda + (lminb0p - lbda)*fj
1301
1302        # final basic transmission loss not exceeded p% of the time
1303        [-5.0*log10(10.0**(-0.2*lbs)+10**(-0.2*lbam)),aht,ahr]
1304      end
1305
1306      # Generic Statistical Estimation of Terrain Diffraction Loss
1307      #
1308      # From:
1309      #
1310      # G. Y. Delisle, J. P. Lefevre, M. Lecours, and J.Y. Choinard. Propagation
1311      # Loss Prediction: A Comparative Study with Application to the Mobile Radio
1312      # Channel. IEEE Trans on Vehicular Technology. Vol. VT-34. No. 2. May, 1985.
1313      #
1314      # f is carrier in MHz
1315      # delta_h is a terrain roughness parameter which might be somewhere in the
1316      #   neighborhood of 15.0 for open terrain, 200ish for hilly terrain, and
1317      #   400ish for rugged terrain
1318      def terrain_diffraction_estimate(f,delta_h=15.0)
1319        r = distance(self.tx,self.rx)
1320        hb = self.tx.h
1321        hm = self.rx.h
1322
```

```
1323      # effective heights in m
1324      heb = hb # I'm not sure how this differs from heights
1325      # Wikipedia seems to imply they are the same:
1326      # http://en.wikipedia.org/wiki/Effective_height
1327      hem = hm
1328
1329      # horizon distances in m
1330      dlsb = sqrt(17.0*heb)
1331      dlsm = sqrt(17.0*hem)
1332
1333      a = Proc.new { |v|
1334        (v > 2.4) ? 12.953 + 20.0*log10(v) : 6.02 + 9.11*v - 1.27*(v**2)
1335      }
1336      dhr = Proc.new { |dh,r|
1337        dh*(1.0 - 0.8*exp(-0.02*r))
1338      }
1339
1340      dlb = dlsb*exp(-0.07*sqrt(delta_h/[5.0,heb].max))
1341      dlm = dlsm*exp(-0.07*sqrt(delta_h/[5.0,hem].max))
1342
1343      dl = dlb + dlm
1344      dls = dlsb + dlsm
1345
1346      theta_eb = (0.0005/dlsb)*(1.3*((dlsb/dlb)-1.0)*delta_h - 4.0*heb)
1347      theta_em = (0.0005/dlsm)*(1.3*((dlsm/dlm)-1.0)*delta_h - 4.0*hem)
1348
1349      d1prime = dl + 0.5*((72165000.0/f)**(1.0/3.0))
1350      d1 = (d1prime <= dls) ? dls : d1prime
1351      d2 = d1 + ((72165000.0/f)**(1.0/3.0))
1352
1353      theta1 = [theta_eb+theta_em,-dl/8495.0].max + d1/8495.0
1354      theta2 = [theta_eb+theta_em,-dl/8495.0].max + d2/8495.0
1355
1356      vb1 = 1.2915*theta1*sqrt(f*dlb*(d1-dl)/(d1-dlm))
1357      vb2 = 1.2915*theta2*sqrt(f*dlb*(d2-dl)/(d1-dlm))
1358      vm1 = 1.2915*theta1*sqrt(f*dlm*(d1-dl)/(d1-dlb))
1359      vm2 = 1.2915*theta2*sqrt(f*dlm*(d2-dl)/(d1-dlb))
1360
1361      ak1 = a.call(vb1) + a.call(vm1)
1362      ak2 = a.call(vb1) + a.call(vm2)
1363
1364      md = (ak2 - ak1)/(d2 - d1)
1365
1366      sigma = 0.78*dhr.call(delta_h,dls)*exp(-0.5*(dhr.call(delta_h,dls)**(1.0/4.0)))
1367      af0prime = 5.0*log10(1.0 + 0.0001*hm*hb*f*sigma)
1368      af0 = [af0prime,15.0].min
1369      a0 = af0 + ak2 - md*d2
1370
1371      ld = md*r + a0
1372      return [ld]
1373    end
1374
1375    ################ STOCHASTIC MODELS ########################
1376
1377    # The Directional Gain Reduction Factor from:
1378    #
1379    # Greenstein and Erceg. "Gain Reductions Due to Scatter on Wireless
1380    # Paths with Directional Antennas". IEEE Comms. Letters. 1999.
1381    #
1382    # A correction for multipath effects at the receiver due to the receiver
1383    # using a directional antenna.
1384    #
1385    # If vary is false, the median case is given.
1386    #
1387    # For 1.9 GHz
1388
1389    def gain_reduction_factor(f,winter=true,vary=false)
```

```
1390      h2 = self.rx.h
1391      beamwidth = rx.beamwidth
1392      model_name = "Directional Gain Reduction Factor"
1393      @warn.push "Receiver height (#{h2}m) is outside the model's coverage"
1394        if h2 > 10 or h2 < 3
1395      @warn.push "Beamwidth (#{beamwidth} degrees) is outside the model's coverage"
1396        if beamwidth < 17 or beamwidth > 65
1397      @warn.push "Frequency (#{f} MHz) is outside the #{model_name} model's coverage"
1398        if f != 1900.0
1399
1400      return [0.0] if beamwidth == 360
1401
1402      i = (winter) ? 1.0 : -1.0
1403      mu = -(0.53 + 0.1*i)*log(beamwidth/360.0) + (0.50 + 0.04*i)*(log(beamwidth/360.0)**2.0)
1404      sigma = -(0.93 + 0.02*i)*log(beamwidth/360.0)
1405      return [vary ? rnorm(mu,sigma) : mu]
1406    end
1407
1408    # EDAM "directivity" model from:
1409    #
1410    # Eric Anderson, Gary Yee, Caleb Phillips, Douglas Sicker, and Dirk Grunwald.
1411    # The Impact of Directional Antenna Models on Simulation Accuracy. 7th
1412    # International Symposium on Modeling and Optimization in Mobile, Ad Hoc,
1413    # and Wireless Networks (WiOpt 2009). Seoul, Korea. June 23 - 27, 2009.
1414    #
1415    # If vary is false, the median case is given.
1416    #
1417    # For 2.4GHz
1418    def edam(f,config,environment=:open_outdoor,vary=false)
1419      @warn.push "Frequency #{f} MHz is out of range for EDAM's coverage"
1420        if f > 2500.0 or f < 2400.0
1421
1422      # setup ranges
1423      kgain = nil
1424      soff = nil
1425      sss = nil
1426      case environment
1427        when :open_outdoor
1428          kgain = [0.01,0.04]
1429          soff = [1.326,2.675]
1430          sss = [2.68,3.75]
1431        when :urban_outdoor
1432          kgain = [0.15,0.19]
1433          soff = [2.244,3.023]
1434          sss = [2.46,2.75]
1435        when :los_indoor
1436          kgain = [0.25,0.38]
1437          soff = [2.837,5.242]
1438          sss = [2.9,5.28]
1439        when :nlos_indoor
1440          kgain = [0.67,0.70]
1441          soff = [3.17,3.566]
1442          sss = [3.67,6.69]
1443      end
1444
1445      # select uniformly at random from within range
1446      kgain = vary ? runif(kgain[0],kgain[1]) : kgain.mean
1447      soff = vary ? runif(soff[0],soff[1]) : soff.mean
1448      sss = vary ? runif(sss[0],sss[1]) : sss.mean
1449
1450      f_src = ant_gain(self.tx,self.rx,config)
1451      f_dst = ant_gain(self.rx,self.tx,config)
1452
1453      g_src = (f_src*kgain + (vary ? rnorm(0.0,soff) : 0.0))
1454      g_dst = (f_dst*kgain + (vary ? rnorm(0.0,soff) : 0.0))
1455
1456      epsilon = (vary) ? rnorm(0.0,sss) : 0.0
```

```
1457
1458      return [g_src,g_dst,epsilon]
1459    end
1460
1461
1462    # Herring Air-to-Ground Model
1463    #
1464    # From: Keith Herring, Jack Holloway, David Staelin. "Path-Loss Characteristics
1465    # of Urban Wireless Channels". IEEE Trans. On Antennas and Propogation. 2009
1466    #
1467    # This is a stochastic measurement-based predictor for 2.4GHz
1468    def herring_atg(f,vary=false)
1469      [freespace(f,2.0).sum,(vary ? rnorm(30,8.3) : 30.0)]
1470    end
1471
1472    # Herring Ground-to-Ground Model
1473    #
1474    # Assumes a single corner between two radios at street level.
1475    #
1476    # This is a stochastic measurement-based predictor for 2.4GHz
1477    def herring_gtg(f,vary=false)
1478      alpha = vary ? runif(2.0,5.0) : [2.0,5.0].mean
1479      ahat = alpha + (vary ? rnorm(0.0,0.22) : 0.0)
1480      b = vary ? rnorm(40.0,5.5) : 40.0
1481      [freespace(f,ahat).sum,b]
1482    end
1483
1484    # TM-90 Model
1485    #
1486    # From:
1487    #
1488    # William Daniel and Harry Wong. Propagation in Suburban Areas at Distances
1489    # less than Ten Miles. FCC Technical Report. FCC/OET TM 91-1. January 25, 1991.
1490    #
1491    def tm90(f,eirp,building_penetration=false)
1492      dkm = distance(self.tx,self.rx)
1493      d = dkm*3280.84 # feet
1494      h1 = self.tx.h*3.28 # feet
1495      h2 = self.tx.h*3.28 # feet
1496      b = building_penetration ? -5.75 + 4.5*log(f) : 0.0
1497      bigf = 141.4 + 20.0*log10(h1*h2) - 40.0*log(d) + b
1498      # Now attempt to convert this value, which is in dBuV/m to dB
1499      # I'm using here, the same equations that SPLAT! uses, but
1500      # I'm not sure where they came from...
1501      erp = eirp - 2.14
1502      p = 10**(erp/10)/1000.0
1503      ldb = 10*log10(p/1000.0) + 139.4 + 20*log10(f) - bigf
1504      [10*log10(p/1000.0),139.4,20*log10(f),-bigf]
1505    end
1506
1507    # IMT-2000 Pedestrian Environment Model
1508    #
1509    # From: Vikay J. Garg. Wireless Communications and Networking. Elsevier. 2007. p. 73.
1510    #
1511    # This is an attempt at worst-case path loss for urban environments, which
1512    # assumes transmitters are outdoors and receivers are indoors. Hence, it
1513    # assumes a outdoor-indoor penetration loss (of 18 dB), a shadowing loss (of 10 dB)
1514    # and a PL exponent of 4.
1515    #
1516    # If vary is false, median case is given
1517    def imt2000_pedestrian(f,indoor_receivers=false,vary=false)
1518      penetration_loss = indoor_receivers ? 18.0 : 0.0
1519      shadowing_loss = vary ? rlognorm(0.0,10.0) : 0.0
1520      d = distance(self.tx,self.rx) # in Km
1521      [40.0*log10(d),30.0*log10(f),shadowing_loss+penetration_loss,21]
1522    end
1523
```

```
1524    # Erceg-Greenstein Model
1525    #
1526    # From: V. Erceg, L. Greenstein, S. Tjandra, S. Parkoff, A. Gupta, B. Kulic,
1527    # A. Julius, and R. Bianchi. An Empirically Based Path Loss Model for Wireless
1528    # Channels in Suburban Environments. Journal on Selected Areas in Communications.
1529    # Vol. 17 No. 7. July, 1999.
1530    #
1531    # For 1.9 GHz
1532    #
1533    # terrain_category can be:
1534    #   :A - Hilly/Moderate to Heavy Tree Density
1535    #   :B - Hilly/Light Tree Density or Flat/Moderate-to-Heavy Tree Density
1536    #   :C - Flat/Light Tree Density
1537    #
1538    # f is the frequency in MHz
1539    # if vary is false, median case is given
1540    def erceg_greenstein(f,terrain_category=:C,vary=false)
1541      # variables
1542      d = distance(self.tx,self.rx)*1000.0 # m
1543      d0 = 100.0 # reference distance in m
1544      # PL in dB at reference dist for this freq
1545      biga = freespace(f,2.0,d0/1000.0).sum
1546      hb = self.rx.h
1547
1548      # static model params
1549      a =           {:A => 4.6,    :B => 4.0,    :C => 3.6}
1550      b =           {:A => 0.0075, :B => 0.0065, :C => 0.0050}
1551      c =           {:A => 12.6,   :B => 17.1,   :C => 20.0}
1552      sigma_gamma = {:A => 0.57,   :B => 0.75,   :C => 0.59}
1553      mu_sigma =    {:A => 10.6,   :B => 9.6,    :C => 8.2}
1554      sigma_sigma = {:A => 2.3,    :B => 3.0,    :C => 1.6}
1555
1556      # pick the right params for the terrain
1557      a = a[terrain_category]
1558      b = b[terrain_category]
1559      c = c[terrain_category]
1560      sigma_sigma = sigma_sigma[terrain_category]
1561      sigma_gamma = sigma_gamma[terrain_category]
1562      mu_sigma = mu_sigma[terrain_category]
1563
1564      # three zero-mean unit standard-deviation gaussian random vars
1565      # x*sigma_gamma is truncated at +/- 1.5
1566      # y and z are truncated at +/- 2.0
1567      # in order to avoid impossible values (however unlikely)
1568      x = [[(vary ? rnorm(0.0,1.0) : 0.0)*sigma_gamma,1.5].min,-1.5].max
1569      # truncate these two to make sure
1570      y = [[(vary ? rnorm(0.0,1.0) : 0.0),2.0].min,-2.0].max
1571      z = [[(vary ? rnorm(0.0,1.0) : 0.0),2.0].min,-2.0].max
1572
1573      return [biga,10*(a - b*hb + c/hb)*log10(d/d0),10.0*x*log10(d/d0) +
1574        y*mu_sigma + y*z*sigma_sigma]
1575    end
1576
1577    # Barclay-Okumura Fading
1578    #
1579    # Frequency-dependent fading based on data from Okumura and several
1580    # other publications, included in:
1581    #
1582    # Les Barclay. Propagation of Radiowaves. IEE. 2003. p. 209
1583    #
1584    # Environment can be either :urban or :suburban
1585    # if vary is false, returns median case which is always zero
1586    def okumura_fc(f,environment=:urban,vary=false)
1587      a = environment == :urban ? 5.2 : 6.6
1588      sigma = 0.65*log10(f)**2 - 1.3*log10(f) + a
1589      [vary ? rnorm(0.0,sigma) : 0.0]
1590    end
```

```
1591 end
```

C.3 Effective Signal to Noise Ratio

Following is an implementation, in R, of the Effective SNR calculation used in this thesis. Some of the functions were derived from the Matlab implementation of Halperin *et al.* in [90]. To conserve space some of the simpler supporting functions have been excluded.

```
1  # 2.1-98 in Proakis
2  Q <- function(x){
3    0.5*erfc(x/sqrt(2))
4  }
5
6  Qinv <- function(y){
7    sqrt(2)*erfcinv(2*y)
8  }
9
10 # Marcum Q from 2.1-122 in Proakis
11 Q1 <- function(a,b,kmax=100){
12   s <- 0
13   for(k in seq(0,kmax)){
14     s <- s + ((a/b)^k)*besselI(a*b,k)
15   }
16   exp(-(a^2 + b^2)/2)*s
17 }
18
19 # ps is probability of symbol error, which
20 # is mod dependent
21
22 # 5.2-57 in Proakis
23 ps.bpsk <- function(snr){
24   Q(sqrt(2*snr))
25 }
26
27 ps.bpsk.inv <- function(ber){
28   (Qinv(ber)^2)/2
29 }
30
31 # 5.2-59 in Proakis
32 ps.qpsk <- function(snr){
33   2*Q(sqrt(2*snr))*(1 - 0.5*Q(sqrt(2*snr)))
34 }
35
36 # inverse solution via the quadratic equation...
37 ps.qpsk.inv <- function(ber){
38   a <- (Qinv(1-sqrt(1-ber))^2)/2
39   b <- (Qinv(1+sqrt(1-ber))^2)/2
40   if(is.finite(a) & is.finite(b)){
41     c(a,b)
42   }else if(is.finite(a)){
43     c(a)
44   }else if(is.finite(b)){
45     c(b)
46   }else{
47     NA
48   }
```

```
49  }
50
51  # A simpler version due to Daniel Halperin <dhalperi@cs.washington.edu>
52  # linux-80211n-csitool-supplementary/matlab/qpsk_berinv.m
53  ps.qpsk.inv.dh <- function(ber){
54    Qinv(ber)^2
55  }
56
57  # 5.2-61 in Proakis
58  ps.mpsk <- function(snr,m){
59    2*Q(sqrt(2*snr)*sin(pi/m))
60  }
61
62  ps.mpsk.inv <- function(ber,m){
63    0.5*(Qinv(ber/2)/sin(pi/m))^2
64  }
65
66  # 5.2-78 in Proakis
67  ps.sqmqam <- function(snr,m){
68    2*(1 - (1/sqrt(m)))*Q(sqrt((3/(m-1))*snr))
69  }
70
71  ps.sqmqam.inv <- function(ber,m){
72    (Qinv(ber/(2*(1-(1/sqrt(m)))))*(m-1))/3
73  }
74
75  # 5.2-79 in Proakis
76  ps.mqam <- function(snr,m){
77    1 - (1 - ps.sqmqam(snr,sqrt(m)))^2
78  }
79
80  ps.mqam.inv <- function(ber,m){
81    ps.sqmqam.inv(1 - sqrt(1 - ber),sqrt(m))
82  }
83
84  # These four via Daniel Halperin <dhalperi@cs.washington.edu>
85  # linux-80211n-csitool-supplementary/matlab/...
86  ps.16qam.inv <- function(ber){
87    Qinv(ber*4/3)^2 * 5
88  }
89  ps.64qam.inv <- function(ber){
90    Qinv(12/7*ber)^2 * 21
91  }
92  ps.16qam <- function(snr){
93    3/4 * Q(sqrt(snr/5))
94  }
95  ps.64qam <- function(snr){
96    7/12 * Q(sqrt(snr/21))
97  }
98  ps.qpsk.dh <- function(snr){
99    Q(sqrt(snr))
100 }
101
102 bits.per.sym <- function(mod){
103   if(mod == "qpsk") 2
104   else if(mod == "dbpsk") 1
105   else if(mod == "dqpsk") 2
106   else if(mod == "bpsk") 1
107   else if(mod == "qam16") 4
108   else if(mod == "qam64") 6
109 }
110
111 # 5.2-70 in Proakis
112 pb.dqpsk <- function(snr){
113   a <- sqrt(2*snr*(1 - sqrt(1/2)))
114   b <- sqrt(2*snr*(1 + sqrt(1/2)))
115   Q1(a,b) - (1/2)*besselI(a*b,0)*exp((-1/2)*(a^2 + b^2))
```

```
116 }
117
118 # 5.2-69 in Proakis
119 pb.dbpsk <- function(snr){
120   (1/2)*exp(-snr)
121 }
122
123 # NOTE: I've "turned on" David Halperin's
124 #       alternative versions of several functions below
125 #       his versions deviate from Proakis and are
126 #       simpler (probably approximations), but are easier
127 #       to compute, invert, and are comparable with the
128 #       Effective SNR paper.
129
130 # pb is probabilility of bit error: (1/j)*ps where
131 # j is the number of bits per symbol (which is mod dep)
132 # 5.2-62 in Proakis
133 pb <- function(snr,mod){
134   j <- bits.per.sym(mod)
135   if(mod == "bpsk") (1/j)*ps.bpsk(snr)
136   else if(mod == "qpsk") (1/j)*ps.qpsk.dh(snr)
137 #  else if(mod == "qpsk") (1/j)*ps.qpsk(snr)
138   else if(mod == "qam16") (1/j)*ps.16qam(snr)
139   else if(mod == "qam64") (1/j)*ps.64qam(snr)
140 #  else if(mod == "qam16") (1/j)*ps.mqam(snr,16)
141 #  else if(mod == "qam64") (1/j)*ps.mqam(snr,64)
142   else if(mod == "dbpsk") pb.dbpsk(snr)
143   else if(mod == "dqpsk") pb.dqpsk(snr)
144 }
145
146 pb.inv <- function(ber,mod){
147   j <- bits.per.sym(mod)
148   if(mod == "bpsk") ps.bpsk.inv(ber*j)
149   else if(mod == "qpsk") ps.qpsk.inv.dh(ber*j)
150 # else if(mod == "qpsk") ps.qpsk.inv(ber*j)
151   else if(mod == "qam16") ps.16qam.inv(ber*j)
152   else if(mod == "qam64") ps.64qam.inv(ber*j)
153 # else if(mod == "qam16") ps.mqam.inv(ber*j,16)
154 # else if(mod == "qam64") ps.mqam.inv(ber*j,64)
155
156 }
157
158 # From: http://msenux.redwoods.edu/math/R/StandardNormal.php
159 stand.norm <- function(x){
160   1/sqrt(2*pi)*exp(-x^2/2)
161 }
162
163 # From Pursley \textit{et al.} Properties and Performance of the IEEE 802.11b
164 # Complementary-Code-Key Signal Sets. IEEE Trans on Comms. Feb. 2009.
165 pu.cck <- function(snr,k,l2=8){
166   n <- k/2
167   i <- c()
168   # fake vectorization
169   b = sqrt(2*snr)
170   if(length(b) > 1){
171     for(bprime in b){
172       # eq. 18
173       integrand <- function(x){ ((2*stand.norm(x+bprime)-1)^(n-1)) *
174         (exp((-x^2)/2)/sqrt(2*pi)) }
175       i <- append(i,integrate(integrand,lower=-bprime,upper=Inf)$value)
176     }
177   }else{
178     integrand <- function(x){ ((2*stand.norm(x+b)-1)^(n-1)) *
179       (exp((-x^2)/2)/sqrt(2*pi)) }
180     i <- integrate(integrand,lower=-b,upper=Inf)$value
181   }
182   pe2 <- 1 - i
```

```
183
184     # eq. 20
185     1 - (1 - pe2)^l2
186   }
187
188   # pu is probability of uncorrectable symbol error
189   # cr is coding rate, k is number of subcarriers
190   # k is 1 for non OFDM and usually 52 for 802.11-style OFDM
191   pu.mod <- function(snr,mod,cr,k){
192     if(mod == "cck16") pu.cck(snr,16)
193     else if(mod == "cck256") pu.cck(snr,256)
194     else{
195       j <- bits.per.sym(mod)
196       m <- k*j # number of total bits
197       t <- 0   # number of correctable bits
198       if(cr == (1/2)) t <- 4
199       else if(cr == (2/3)) t <- 2
200       else if(cr == (3/4)) t <- 2
201       else if(cr == 1) t <- 0
202
203       if(consider.coding && (t > 0)) choose(m,t+1)*(pb(snr,mod)^(t+1))
204       else pb(snr,mod)
205     }
206   }
207
208   pu.mod.inv <- function(ber,mod,cr,k){
209     j <- bits.per.sym(mod)
210     m <- k*j # number of total bits
211     t <- 0   # number of correctable bits
212     if(cr == (1/2)) t <- 4
213     else if(cr == (2/3)) t <- 2
214     else if(cr == (3/4)) t <- 2
215     else if(cr == 1) t <- 0
216
217     if(consider.coding && (t > 0)) pb.inv((ber/choose(m,t+1))^(1/(t+1)),mod)
218     else pb.inv(ber,mod)
219   }
220
221   pu <- function(snr,rate){
222     ofdm.k <- 52 # 48 + 4 pilots
223     cr <- NULL   # coding rate
224     k <- NULL    # number of subcarriers
225     mod <- NULL
226
227     # values from 802.11 spec table 17-3
228     if(rate == 1){
229       mod <- "dbpsk"
230       k <- 1
231       cr <- 1
232     }else if(rate == 2){
233       mod <- "dqpsk"
234       cr <- 1
235       k <- 1
236     }else if(rate == 6){
237       mod <- "bpsk"
238       k <- ofdm.k
239       cr <- 1/2
240     }else if(rate == 9){
241       mod <- "bpsk"
242       cr <- 3/4
243       k <- ofdm.k
244     }else if(rate == 12){
245       mod <- "qpsk"
246       cr <- 1/2
247       k <- ofdm.k
248     }else if(rate == 18){
249       mod <- "qpsk"
```

```
250       cr <- 3/4
251       k <- ofdm.k
252     }else if(rate == 24){
253       mod <- "qam16"
254       cr <- 1/2
255       k <- ofdm.k
256     }else if(rate == 36){
257       mod <- "qam16"
258       cr <- 3/4
259       k <- ofdm.k
260     }else if(rate == 48){
261       mod <- "qam64"
262       cr <- 2/3
263       k <- ofdm.k
264     }else if(rate == 54){
265       mod <- "qam64"
266       cr <- 3/4
267       k <- ofdm.k
268     }else if(rate == 11){
269       mod <- "cck256"
270       cr <- 1
271       k <- 1
272     }else if(rate == 5){
273       mod <- "cck16"
274       cr <-1
275       k <- 1
276     }
277     pu.mod(snr,mod,cr,k)
278   }
279
280   pu.inv <- function(ber,rate){
281     ofdm.k <- 52 # 48 + 4 pilots
282     cr <- NULL    # coding rate
283     k <- NULL     # number of subcarriers
284     mod <- NULL
285
286     # values from 802.11 spec table 17-3
287     if(rate == 6){
288       mod <- "bpsk"
289       k <- ofdm.k
290       cr <- 1/2
291     }else if(rate == 9){
292       mod <- "bpsk"
293       cr <- 3/4
294       k <- ofdm.k
295     }else if(rate == 12){
296       mod <- "qpsk"
297       cr <- 1/2
298       k <- ofdm.k
299     }else if(rate == 18){
300       mod <- "qpsk"
301       cr <- 3/4
302       k <- ofdm.k
303     }else if(rate == 24){
304       mod <- "qam16"
305       cr <- 1/2
306       k <- ofdm.k
307     }else if(rate == 36){
308       mod <- "qam16"
309       cr <- 3/4
310       k <- ofdm.k
311     }else if(rate == 48){
312       mod <- "qam64"
313       cr <- 2/3
314       k <- ofdm.k
315     }else if(rate == 54){
316       mod <- "qam64"
```

```
317       cr <- 3/4
318       k <- ofdm.k
319     }
320     pu.mod.inv(snr,mod,cr,k)
321   }
322
323   erate <- function(snr,rate){
324     rate*(1 - pu(snr,rate))
325   }
326
327   # Receiver minimum input sensitivity from the 802.11 spec Table 17-13
328   rmis <- function(r){
329     if(r == 1) -85
330     else if(r == 2) -84
331     else if(r == 6) -82
332     else if(r == 12) -80
333     else if(r == 24) -77
334     else if(r == 36) -73
335     else if(r == 48) -69
336     else if(r == 54) -68
337     else 0
338   }
```

C.4 Spatial Simulated Annealing

The following R code performs spatial simulated annealing. It uses the code listed in C.1 and the roughness function that follows next. It assumes there is a list of candidate sample locations named "candidates" from which measurements locations are selected.

```
1   # simulated annealing
2   n <- 50
3   tmax <- 2000
4   dcoords.new <- NULL
5   num.children <- 12
6   parallelize <- TRUE
7
8   # first argument is a period-separated list of indices into the
9   # candidates dataframe
10  e <- commandArgs(TRUE)[1]
11  e <- as.numeric(unlist(strsplit(e, "\\.")))
12  de <- candidates[e,]
13  runid <- commandArgs(TRUE)[2]
14  e_before <- e
15
16  t <- tmax
17  if(parallelize){
18    c1 <- makeForkCluster(num.children)
19    kv <- krige.var.par(rbind(dcoords,de[,c("x","y")]),loci,kc,c1)
20  }else{
21    kv <- krige.var(rbind(dcoords,de[,c("x","y")]),loci,kc)
22  }
23  vmap <- flipud(matrix(kv,nrow=height,ncol=width,byrow=TRUE))
24  fitness <- wpe(rmap,vmap)
25  fitness2 <- mean(sqrt(vmap))
26  first.fitness <- fitness
27  first.fitness2 <- fitness2
28  rm(kv,vmap)
29
```

```
30  linear.cooling = FALSE
31
32  log <- NULL
33
34  while(t > 0){
35    e2 <- e[sample(seq(1,n),n-1)] # n - 1 sized sample of indices
36    de2 <- candidates[e2,]
37    while(length(e2) < n){
38      p <- sample(seq(1,nrow(candidates)),1)
39      if(any(e2 == p)) next
40      e2 <- append(e2,p)
41      de2 <- rbind(de2,candidates[p,])
42    }
43    if(parallelize){
44      kv <- krige.var.par(rbind(dcoords,de2[,c("x","y")]),loci,kc,c1)
45    }else{
46      kv <- krige.var(rbind(dcoords,de2[,c("x","y")]),loci,kc)
47    }
48    vmap <- flipud(matrix(kv,nrow=height,ncol=width,byrow=TRUE))
49    new.fitness <- wpe(rmap,vmap)
50    new.fitness2 <- mean(sqrt(vmap))
51    rm(vmap,kv)
52
53    replaced <- TRUE
54    deltaf <- new.fitness-fitness
55    p <- NA
56    if(deltaf < 0){
57      fitness <- new.fitness
58      fitness2 <- new.fitness2
59      e <- e2
60      de <- de2
61    }else{
62      if(linear.cooling){
63        p <- t/tmax
64      }else{
65        temp <- t/tmax
66        # scale up deltaf by 10^2 to get a more meaningful cooling curve
67        p <- exp(-100.0*deltaf/temp)
68      }
69      print(paste(t,"worse :(",fitness,fitness2,deltaf,p))
70      if(runif(1) <= p){
71        print("accepted badness")
72        fitness <- new.fitness
73        fitness2 <- new.fitness2
74        e <- e2
75        de <- de2
76      }else{
77        replaced <- FALSE
78      }
79    }
80    t <- t - 1
81    log <- rbind(log,data.frame(t=t,replaced=replaced,p=p,fitness=fitness,
82                 fitness2=fitness2,deltaf=deltaf))
83  }
84
85  if(parallelize){
86    stopCluster(c1)
87  }
88
89  wpe.gain <- first.fitness-fitness
90  akv.gain <- first.fitness2-fitness2
91
92  etime <- as.numeric(Sys.time())
93
94  print(cat("FITNESS",first.fitness,fitness,first.fitness2,fitness2,etime,""))
95  print(cat("SAMPLE",e,""))
96
```

```
97  save(n, first.fitness, first.fitness2, dcoords, wpe.gain, akv.gain, fitness,
98       fitness2, etime, e_before, e, log, tmax, candidates,
99       file=paste(sep="","sa_slave_",runid,"_",etime,".RData"))
100
101 print(cat("DONE",""))
```

The following code computes the roughness map given a map (as a matrix). A function to compute the WPE using this roughness map and the kriging variance map are also provided.

```
1   # calculate roughness
2   roughness <- function(map, height, width, nr=1, pix.per.m=0.2, beta=1.5, alpha=1.0){
3     height <- nrow(map)
4     width <- ncol(map)
5     ret <- map
6     neigh <- expand.grid(seq(-nr, nr), seq(-nr, nr))
7     dsum <- 0.0
8
9     for(k in nrow(neigh)){
10      x <- neigh[k,1]
11      y <- neigh[k,2]
12      if(x == 0 && y == 0) next;
13      d <- sqrt(x^2 + y^2)/pix.per.m
14      dsum <- dsum + d
15    }
16    for(i in seq(1,height)){
17      for(j in seq(1,width)){
18        s <- 0.0
19        v <- map[i,j]
20        for(k in nrow(neigh)){
21          x <- neigh[k,1]
22          y <- neigh[k,2]
23          xi <- x + j
24          yi <- y + i
25          if(x == 0 && y == 0) next;
26          if(xi < 1 || yi < 1 || xi > width || yi > height) next;
27          d <- sqrt(x^2 + y^2)/pix.per.m
28          v2 <- map[yi,xi]
29          s <- s + (d^(-beta) * (v2 - v)^2)/dsum
30        }
31        ret[i,j] <- s
32      }
33    }
34    ret <- (ret/max(ret))^alpha
35    return(ret)
36  }
37
38  wpe <- function(rmap,vmap){
39    # note this is not a matrix mult (%*%) so will just multiple rmap[i,i]*vmap[i,i]
40    mean(rmap*vmap)
41  }
```

C.5 Variogram Fitting and Kriging

A simplified (but still largely complete) version of the variogram fitting and kriging code, utilizing the geoR library, is provided below.

```
1  library(geoR) # for all the kriging stuff
2  library(lattice) # xyplot and friends
3  library(dichromat) # for ramp()
4  library(matlab) # for flipud()
5
6  guess.range <- function(v){
7    lastval <- 0
8    thisi <- 1
9    for(i in 1:length(v$v)){
10     thisval <- v$v[i]
11     if(thisval < lastval) break;
12     thisi <- i
13     lastval <- thisval
14   }
15   v$u[i]
16 }
17
18 krige.per.ap <- function(fname,subtitle,fnsubtitle,lag,ap,nug.tol,d,metric,freq,
19                          noiseval,max.dist.m,na.value,p.tx,data.combined=FALSE){
20
21   minval = min(d$sig,na.rm=TRUE)
22   maxval = max(d$sig,na.rm=TRUE)
23   valrange = maxval-minval
24
25   d2 <- d
26   d2$non <- FALSE
27   d2[is.na(d2$sig),"non"] <- TRUE
28   d2[is.na(d2$sig),"sig"] <- na.value # d2 has NA replaced with na.value
29
30   # convert signal to total PL (if possible)
31   if(metric == "snr" || metric == "esnr6" || metric == "esnr54"){
32     # SNR = P_tx - (N + PL) = P_tx - N - PL
33     # t.f. PL = P - N - SNR
34     d2$sig <- p.tx - noiseval - d2$sig
35     print(paste("NA Value in PL = ",p.tx - noiseval - na.value,
36       "Versus minimum PL observed = ",min(d2$sig)))
37     metric <- "pl"
38   }else if(metric == "rss"){
39     d2$sig <- p.tx - d2$sig
40     metric <- "pl"
41   }else if(metric == "tput"){
42     d2$sig <- (d2$sig-minval)/valrange
43   }
44
45   sigma2 <- NA
46   if(metric == "pl" && !all(is.na(d$dist))){
47     # Fit friis' PL to data
48     m2 <- lm(sig ~ log10(dist),data=d2)
49
50     print("Model Fitting Summary")
51     print(summary(m2))
52
53     slope2 <- m2$coefficients[2]
54     intercept2 <- m2$coefficients[1]
55     alpha2 <- slope2/10
56     epsilon2 <- intercept2 - 20*log10(freq) - 32.45
57
58     sigma2 <- round(summary(m2)$sigma,3)
59
60     # sig2 is PL reduced by frii's trivial PL
61     d2$sig2 <- d2$sig - friis(d2$dist,freq,alpha2,epsilon2)
62   }else{
63     # Don't know how to remove trend for other metrics, so just
64     # do nothing...
65     d2$sig2 <- d2$sig
66   }
67
```

```
68    n <- nrow(d2)
69
70    # d1 has null measurements excluded
71    d1 <- d2[!d2$non,]
72
73    d1coords <- data.frame(x=d1$east,y=d1$north)
74    d2coords <- data.frame(x=d2$east,y=d2$north)
75
76    # if we are combing across several APs, co-located points must be jittered
77    if(data.combined){
78      print("jittering duplicated coordinates by up to 20 wavelengths")
79      # jitter up to 20 wavelengths
80      maxjitter <- 2*freq.to.wavelength(freq/1000.0)*20.0
81      # is it bad that these are being independently jittered?
82      d2coords <- jitter2d(d2coords,max=maxjitter)
83      d1coords <- jitter2d(d1coords,max=maxjitter)
84    }
85
86    eastrng <- range(d2coords$x)
87    northrng <- range(d2coords$y)
88
89    # representative example of detrended, truncated, and with null measurements
90    # given, although this may not be the best performing model for all scenarios
91
92    # compute empirical variogram
93    v2.detrend.trunc <- variog(coords=d2coords,data=d2$sig2,
94      nugget.tolerance=nug.tol,option="bin",max.dist=max.dist.m)
95
96    # perform fitting
97    range.ini <- guess.range(v2.detrend.trunc)
98    nug.ini <- v2.detrend.trunc$v[1]
99    sill.ini <- max(v2.detrend.trunc$v)-nug.ini
100   v2.detrend.trunc.fit.gauss <- variofit(v2.detrend.trunc,cov.model="gaussian",
101     ini.cov.pars=c(sill.ini,range.ini),nugget=nug.ini,fix.nugget=TRUE)
102   v2.detrend.trunc.fit.cubic <- variofit(v2.detrend.trunc,cov.model="cubic",
103     ini.cov.pars=c(sill.ini,range.ini),nugget=nug.ini,fix.nugget=TRUE)
104
105   n.sample.max <- 50   # max points to validate
106   n.sample.frac <- 0.2 # fraction of points to validate
107   n.folds <- 10
108   n.sample <- min(c(n.sample.max, ceil(n.sample.frac*length(d2$sig2))))
109
110   # try both cubic and gaussian fits and keep whichever is better
111   best.model <- NA
112   best.model.truncated <- TRUE
113   best.model.rmse <- NA
114   best.model.name <- NA
115   best.model.non <- NA
116
117   v <- do.validate(d2,d2coords,v2.detrend.fit.gauss,n.sample,n.folds)
118   best.model <- v2.detrend.fit.gauss
119   best.model.truncated <- FALSE
120   best.model.rmse <- mean(v$rmse)
121   best.model.name <- "Gaussian w/ Null"
122   best.model.non <- TRUE
123   fitstats <- rbind(fitstats,data.frame(m=v2.detrend.fit.gauss$cov.model,
124     ssq=v2.detrend.fit.gauss$value,sigmasq=v2.detrend.fit.gauss$cov.pars[1],
125     phi=v2.detrend.fit.gauss$cov.pars[2],kappa=v2.detrend.fit.gauss$kappa,
126     tausq=v2.detrend.fit.gauss$nugget,ap=ap,wneg=TRUE,truncated=FALSE,
127     lag=lag,n=n,xv.rmse.mean=mean(v$rmse),xv.rmse.std=std(v$rmse),
128     mq90=mean(v$q90),mq75=mean(v$q75),mq100=mean(v$q100),
129     xv.mskv.mean=mean(v$mskv),xv.rmse.std=std(v$mskv),sigma1=sigma1,
130     sigma2=sigma2,sigma3=sigma3))
131
132   v <- do.validate(d2,d2coords,v2.detrend.fit.cubic,n.sample,n.folds)
133   if(mean(v$rmse) < best.model.rmse){
134     best.model <- v2.detrend.fit.gauss
```

```
135       best.model.truncated <- FALSE
136       best.model.rmse <- mean(v$rmse)
137       best.model.name <- "Cubic w/ Null"
138       best.model.non <- TRUE
139     }
140     fitstats <- rbind(fitstats,data.frame(m=v2.detrend.fit.cubic$cov.model,
141       ssq=v2.detrend.fit.cubic$value,sigmasq=v2.detrend.fit.cubic$cov.pars[1],
142       phi=v2.detrend.fit.cubic$cov.pars[2],kappa=v2.detrend.fit.cubic$kappa,
143       tausq=v2.detrend.fit.cubic$nugget,ap=ap,wneg=TRUE,truncated=FALSE,lag=lag,
144       n=n,xv.rmse.mean=mean(v$rmse),xv.rmse.std=std(v$rmse),mq90=mean(v$q90),
145       mq75=mean(v$q75),mq100=mean(v$q100),xv.mskv.mean=mean(v$mskv),
146       xv.rmse.std=std(v$mskv),sigma1=sigma1,sigma2=sigma2,sigma3=sigma3))
147
148
149
150     width <- round(diff(range(d2coords$x))*pix.per.meter)
151     height <- round(diff(range(d2coords$y))*pix.per.meter)
152     loci2 <- expand.grid(seq(min(d2coords$x),max(d2coords$x),length.out=width),
153       seq(min(d2coords$y),max(d2coords$y),length.out=height))
154
155     dk <- d2
156     dkcoords <- d2coords
157
158     if(!best.model.non && !best.model.truncated){
159       dk <- d1
160       dkcoords <- d1coords
161     }else if(!best.model.non && best.model.truncated){
162       dk <- d1
163       dkcoords <- d1coords
164     }
165     tryCatch(do.krige(best.model,dk,dkcoords,loci2,"best",best.model.name),
166               error=function(err){ print(paste("ERROR kriging: ",err)); return(0) })
167     print(fitstats)
168     list(fitstats=fitstats,eastrng=eastrng,northrng=northrng,width=width,height=height)
169   }
170
171   do.validate <- function(d2,d2coords,model,n.sample,n.folds){
172     valdata <- NULL
173     for(i in seq(1,n.folds)){
174       tryCatch(x <- xvalid(coords=d2coords,data=d2$sig2,model=model,
175         locations.xvalid=sample(seq(1,length(d2$sig2)),n.sample),
176         error=function(err){ print(paste("ERROR xvalid: ",err)); return(0) })
177       if(length(x) > 1){
178         x.rmse <- sqrt(mean((x$error)^2))
179         x.mskv <- sqrt(mean(x$krige.var))
180         q <- quantile(abs(x$error),probs=c(0.75,0.9,1.0))
181         valdata <- rbind(valdata,data.frame(n=n.sample,f=i,rmse=x.rmse,
182           mskv=x.mskv,q75=q[1],q90=q[2],q100=q[3]))
183       }
184     }
185     valdata
186   }
187
188   do.krige <- function(model,d,dcoords,loci,name,prettyname,local=FALSE,
189     n.local=8,universal=FALSE){
190
191     kc <- krige.control(type.krige="ok",obj.model=model)
192     k <- krige.conv(coords=dcoords,data=d$sig2,locations=loci,krige=kc)
193
194     write.table(flipud(matrix(k$predict,nrow=height,ncol=width,byrow=TRUE)),
195                 file=paste(sep="/",fig.dir,paste(sep="_",ap,fnsubtitle,
196                 eastrng[1],eastrng[2],northrng[1],
197                 northrng[2],pix.per.meter,name,"detrend_map.csv")))
198
199     ape <- d$apeast[1] # all rows should be the same
200     apn <- d$apnorth[1] # ...
201     if(metric == "pl" && !all(is.na(d$dist))){
```

```
202       for(i in seq(1,length(loci[,1]))){
203         e <- loci[i,1]
204         n <- loci[i,2]
205         # distance between grid point and AP in km
206         dist <- sqrt((ape-e)^2 + (apn-n)^2)/1000
207
208         # convert back to signal strength
209         k$predict[i] <- p.tx - (k$predict[i] + friis(dist,freq,alpha2,epsilon2))
210       }
211     }
212
213     print(paste("saving map to file",paste(sep="/",fig.dir,paste(sep="_",ap,
214                   fnsubtitle,eastrng[1],eastrng[2],northrng[1],
215                   northrng[2],pix.per.meter,name,"map.csv"))))
216     write.table(flipud(matrix(k$predict,nrow=height,ncol=width,byrow=TRUE)),
217                   file=paste(sep="/",fig.dir,paste(sep="_",ap,fnsubtitle,eastrng[1],
218                   eastrng[2],northrng[1],
219                   northrng[2],pix.per.meter,name,"map.csv")))
220     write.table(flipud(matrix(k$krige.var,nrow=height,ncol=width,byrow=TRUE)),
221                  file=paste(sep="/",fig.dir,paste(sep="_",ap,fnsubtitle,
222                  eastrng[1],eastrng[2],northrng[1],northrng[2],pix.per.meter,name,
223                  "var_map.csv")))
224
225
226     # make sure everything gets cleaned up
227     k <- NULL
228     gc(verbose=TRUE)
229   }
```

C.6 Anritsu National Instruments Interface

The following C code implements a network-based communication interface to an Anritsu MS2712B portable spectrum analyzer. It was used to partially automate data collection for the experiments described in section 6.1 and 8.

```
1   #include "stdlib.h"
2   #include "stdio.h"
3   #include "unistd.h"
4   #include "string.h"
5   #include "time.h"
6   #include "visa.h"
7
8   #define NO_ERROR 0
9   #define USAGE_ERROR 1
10  #define VISA_ERROR 2
11
12  #define BUFFER_SIZE 512
13
14  #define SIGNAL_STANDARD 9
15  #define CHANNEL_BANDWIDTH 3
16
17  #define DEVICE_TIMEOUT 30
18
19  #define SWITCH_TO_WIMAX 0
20  #define ENABLE_GPS 1
21
22  void usage(){
23    fprintf(stderr,"Usage: ./measure <IP Address> <channel1,channel2,channel3> \
```

```
24  <num measurements>\n");
25     exit(USAGE_ERROR);
26  }
27
28  int do_read_write(ViSession instr, const char *cmd){
29     ViStatus status;
30     ViUInt32 retCount;
31     ViChar vbuffer[BUFFER_SIZE];
32     char cbuffer[BUFFER_SIZE];
33
34     sprintf(vbuffer,"%s",cmd);
35     if((status = viWrite(instr,(unsigned char *)&vbuffer[0],strlen(vbuffer),
36       &retCount)) < VI_SUCCESS){
37
38        viStatusDesc(instr,status,vbuffer);
39        fprintf(stderr,"VISA Write Error: %s\nCommand Was: %s",vbuffer,cmd);
40        return VISA_ERROR;
41     }
42     if((status = viRead(instr,(unsigned char *)vbuffer,BUFFER_SIZE,&retCount)) <
43       VI_SUCCESS){
44
45        viStatusDesc(instr,status,vbuffer);
46        fprintf(stderr,"VISA Read Error: %s\nCommand Was: %s",vbuffer,cmd);
47        return VISA_ERROR;
48     }
49     strncpy(cbuffer,vbuffer,retCount);
50     cbuffer[retCount] = 0; // null terminate
51     printf("%d: %s\n",(int)retCount,cbuffer);
52     return NO_ERROR;
53  }
54
55  int do_write(ViSession instr, const char *cmd, int post_sleep){
56     ViStatus status;
57     ViUInt32 retCount;
58     ViChar vbuffer[BUFFER_SIZE];
59     char cbuffer[BUFFER_SIZE];
60     sprintf(vbuffer,"%s",cmd);
61     if((status = viWrite(instr,(unsigned char *)&vbuffer[0],strlen(vbuffer),
62                          &retCount)) < VI_SUCCESS){
63
64        viStatusDesc(instr,status,vbuffer);
65        fprintf(stderr,"VISA Write Error: %s\nCommand Was: %s",vbuffer,cmd);
66        return VISA_ERROR;
67     }
68     sleep(post_sleep);
69     return NO_ERROR;
70  }
71
72  // http://www.ni.com/pdf/manuals/370132c.pdf
73
74  int main(int argc, char* argv[]){
75     ViStatus status;
76     ViSession defaultRM,instr;
77     ViUInt32 retCount;
78     char cbuffer[BUFFER_SIZE];
79     char tbuffer[BUFFER_SIZE];
80     char *addr, *tok, *channels;
81     time_t rawtime;
82     struct tm *now;
83     int chan, num_measurements;
84
85     if(argc < 4) usage();
86
87     addr = argv[1];
88     channels = argv[2];
89     num_measurements = atoi(argv[3]);
90     status = viOpenDefaultRM(&defaultRM);
```

```
91      if(status < VI_SUCCESS){
92        fprintf(stderr,"Can't initialize VISA\n");
93        return VISA_ERROR;
94      }
95      sprintf(cbuffer,"TCPIP0::%s::INSTR",addr);
96      status = viOpen(defaultRM,cbuffer,VI_NULL,VI_NULL,&instr);
97      status = viSetAttribute(instr,VI_ATTR_TMO_VALUE,DEVICE_TIMEOUT*1000);
98
99      do_read_write(instr,"*IDN?\n");
100
101     if(SWITCH_TO_WIMAX) do_write(instr,":INSTrument:SELect \"WIMAX_E\"\n",30);
102     if(ENABLE_GPS) do_write(instr,":SENSe:GPS ON\n",5);
103
104     tok = strtok(channels,",");
105     while(tok != NULL){
106       chan = atoi(tok);
107       printf("Channel %d\n",chan);
108       printf("Setting Standard (%d), Channel (%d), and Bandwidth (%d)\n",
109         SIGNAL_STANDARD,chan,CHANNEL_BANDWIDTH);
110       sprintf(cbuffer,":SENSe:FREQuency:SIGSTANDARD %d\n",SIGNAL_STANDARD);
111       do_write(instr,(const char *)cbuffer,2);
112       sprintf(cbuffer,":SENSE:FREQUENCY:SIGSTANDARD:CHANNEL %d\n",chan);
113       do_write(instr,(const char *)cbuffer,2);
114       sprintf(cbuffer,":SENSe:BANDwidth %d\n",CHANNEL_BANDWIDTH);
115       do_write(instr,(const char *)cbuffer,2);
116       for(int i = 0; i < num_measurements; i++){
117         fflush(stdout);
118         time(&rawtime);
119         now = localtime(&rawtime);
120         strftime(tbuffer,BUFFER_SIZE,"%Y%m%d%H%M%S",now);
121         printf("Doing measurements %d of %d @ %s\n",i+1,num_measurements,tbuffer);
122         if(ENABLE_GPS) do_read_write(instr,":FETCh:GPS?");
123         printf(" => Configuration\n");
124         sprintf(cbuffer,":MMEMory:STORe:STATe 0,\"con%s\"\n",tbuffer);
125         do_write(instr,(const char *)cbuffer,5);
126         printf(" => Summary\n");
127         do_write(instr,":CONFigure:DEMod SUMMary\n",10);
128         sprintf(cbuffer,":MMEMory:STORe:TRACe 0,\"sum%s\"\n",tbuffer);
129         do_write(instr,(const char *)cbuffer,5);
130         printf(" => Spectrum Flatness\n");
131         do_write(instr,":CONFigure:DEMod SFL\n",10);
132         sprintf(cbuffer,":MMEMory:STORe:TRACe 0,\"sfl%s\"\n",tbuffer);
133         do_write(instr,(const char *)cbuffer,5);
134         printf(" => Constellation Plot\n");
135         do_write(instr,":CONFigure:DEMod CONSTln\n",10);
136         sprintf(cbuffer,":MMEMory:STORe:TRACe 0,\"cns%s\"\n",tbuffer);
137         do_write(instr,(const char *)cbuffer,5);
138         if(ENABLE_GPS) do_read_write(instr,":FETCh:GPS?");
139       }
140       fflush(stdout);
141       tok = strtok(NULL,",");
142     }
143
144     status = viClose(instr);
145     status = viClose(defaultRM);
146
147     return NO_ERROR;
148   }
```

Acronyms

3G Third Generation. 189

3GPP Third Generation Partnership Project. 189

4G Fourth Generation. 189

AIC Akaike Information Criterion. 133

AKV Average Kriging Variance. 258–260, 265

ANOVA Analysis of Variance. 95, 97, 168

AOV Analysis of Variance. 193

AP Access Point. 4, 62, 63, 65, 101, 114, 115, 117, 119, 123, 124, 126, 143, 149, 152, 158, 304

ASCII American Standard Code for Information Interchange. 192

ASN Access Service Network. 168, 171

ATG Air-to-Ground. 34

AWB Accton Wireless Broadband Corporation. 168, 171

BER Bit Error Rate. 139, 140

BLER Block Error Rate. 193

BS Base Station. 4, 105, 106, 156, 162, 164, 168, 173, 178, 180, 184, 186, 189, 192–194, 197, 205, 215, 221, 223, 238, 240, 247, 298

BSSID Basic Service Set Identifier. 67

CAT5 Category 5. 164

CCIR International Radio Consultive Committee. 31

CEPT European Conference of Postal and Telecommunications Administrations. 30

CINR Carrier to Interference and Noise Ratio. 166–168, 171, 173, 178, 180, 186, 254, 267

CIR Channel Impulse Response. 140

CMU Carnegie Mellon University. 4

COST-231 European Cooperation in the field of Scientific and Technical Research Action 231. 34, 51, 52, 58, 69, 77, 93, 94

COTSE Commercial Off The Shelf Equipment. 140, 141, 213, 302

CPE Client Premises Equipment. 62

CPU Central Processing Unit. 274

CQI Channel Quality Indicator. 193, 194

CR Cognitive Radio. 141

CSM Channel State Matrix. 139

CSV Comma Separated Value. 192

CU University of Colorado at Boulder. 63, 67, 105, 107, 156, 162, 184, 186, 189, 192, 210, 232–234, 237, 254, 255, 283, 292

CW Continuous Wave. 57

DEM Digital Elevation Model. 37, 70, 71, 349

DET Detection Error Tradeoff. 224, 240

DSM Digital Surface Model. 53

DTM Digital Terrain Model. 156

ECC Electronic Communication Committee. 30, 52

ECOT Engineering Center Office Tower. 186

EDAM Effective Directivity Antenna Model. 10, 42, 103

EIRP Effective Isotropic Radiated Power. 17, 62, 114, 115, 194, 326

ERSI Environmental Systems Research Institute. 255

ESNR Effective Signal to Noise Ratio. 166, 167, 171, 173, 178, 186, 212, 267

EVM Error Vector Magnitude. 166, 167, 173, 178, 186, 212

FCC Federal Communications Commission. 5, 27, 101, 303

FCS Frame Check Sequence. 58

FDM Finite Difference Method. 104

GENI Global Environment for Networking Innovation. 164, 168, 254

GIS Geographic Information System. 255, 302

GPGPU General Purpose Graphical Processing Unit. 304

GPS Global Positioning System. 6, 62, 65–67, 69, 119, 121, 122, 141, 164, 166, 167, 171, 191, 232, 237

GTG Ground-to-Ground. 34, 35

HTTP Hypertext Transport Protocol. 122

ICMP Internet Control Message Protocol. 122

IDW Inverse Distance Weighting. 137

IEEE Institute of Electrical and Electronics Engineers. 46, 141

ISI Inter-symbol Interference. 44, 140

ITM Irregular Terrain Model. 27, 39, 53, 69, 94, 101, 103, 104, 156, 347–349

ITS Institute for Telecommunications Sciences. 347

ITU International Telecommunications Union. 347

ITU-R International Telecommunications Union Radiocommunication Sector. 31, 38, 39, 53, 70, 93, 94

JDSU JDS Uniphase. 189, 192

KAIST Korea Advanced Institute of Science and Technology. 233

KML Keyhole Markup Language. 156

LCDB Landcover Classification Database. 70, 71

LiDaR Light Detection and Ranging. 105, 106, 109, 304

LOS Line of Sight. 34, 168, 344

LS Least Squares. 132

LTE Long Term Evolution. 10, 158, 162, 189, 192, 193, 197, 212, 213, 215, 240

MAD Median Absolute Deviation. 210

Mbps Megabits per second. 173

MIMO Multiple Input Multiple Output. 139

MIR Multiple Independent Runs. 273

MLE Maximum Likelihood Estimator. 132, 133, 161, 252

MOM Method Of Moments. 161

MSKV Mean Square Kriging Variance. 152, 223, 283, 292, 295

MTU Maximum Transmission Unit. 171

NCALM National Center for Airbourne Laser Mapping. 106

NDA Nondisclosure Agreement. 60

NIST National Institute of Standards and Technology. 34

NLOS No(n) Line of Sight. 22, 34, 168, 344

NSF National Science Foundation. 106

NTIA National Telecommunications and Information Administration. 347

OFDM Orthogonal Frequency Division Multiplexing. 139

OIT Office of Information and Technology. 162

OK Ordinary Kriging. 154, 158, 251, 303

OSM Open Signal Maps. 220, 232, 238, 250, 302

PDSCH Physical Downlink Shared Channel. 193

PL Path Loss. 192–194, 197, 213

POC Proof of Concept. 117, 119, 124, 126

PPM Pixels per Meter. 154

PSK Phase Shift Keying. 167

PU Primary User. 137

PVC Polyvinyl Chloride. 166

QAM Quadrature Amplitude Modulation. 167

RAM Random Access Memory. 274

RCE Relative Constellation Error. 166, 167, 212

REM Radio Environment Map. 140

RF Radio Frequency. 4, 6, 114, 135, 158, 247, 289, 299, 304, 327, 342

RFP Request for Proposals. 117

RMSE Root Mean Square Error. 47, 56, 70, 77, 79, 80, 93–95, 101, 103, 107, 111, 152, 154, 178, 180, 184, 223, 224, 240, 283, 285, 292, 295

ROC Receiver Operating Characteristic. 224, 240

RSRP Reference Signal Received Power. 192

RSS Received Signal Strength. 58, 59, 63, 67, 115, 136, 166, 192, 194, 213, 220

RSSI Received Signal Strength Indicator. 66, 67, 192–194, 327

SA Simulated Annealing. 263, 265, 273, 277

SC-RMSE Spread-Corrected Root Mean Square Error. 79, 80, 93

SCF Subcarrier Flatness. 171

SDR Software Defined Radio. 140

SINR Signal to Interference and Noise Ratio. 17

SLAW Self-similar Least Action Walk. 232, 233

SNR Signal to Noise Ratio. 17, 48, 60, 66, 77, 115, 139–141, 156, 166, 167, 171, 173, 186, 192–194, 213, 223

SRS Sounding Reference Signal. 193

SRS Simple Random Sampling. 113, 116, 117, 135

SSA Spatial Simulated Annealing. 136, 252, 254, 265, 289

STL Stereo Lithography. 105, 106

SUI Stanford University Interim model. 52

TCP Transmit Control Protocol. 189

TFA Technology For All. 48, 65, 67

U-NII Unlicensed National Information Infrastructure. 62

UAV Unmanned Aerial Vehicle. 304

UCSD University of California San Diego. 233

UDP User Datagram Protocol. 168

UE User Equipment. 192–194, 205, 302

UHF Ultra High Frequency. 304

UK United Kingdom. 238

UK Universal Kriging. 158, 303

USB Universal Serial Bus. 65, 69, 121, 122, 213

USGS United States Geological Survey. 255, 349

UTD Uniform Theory of Diffraction. 10, 44, 104, 304

UTM Universal Transverse Mercator. 154

VHF Very High Frequency. 39

VISA Virtual Instrument Software Architecture. 164

VSG Vector Signal Generator. 59

WAND Waikato Applied Network Dynamics. 60

WART Wide Area Radio Testbed. 63, 67

WLS Weighted Least Squares. 132, 137, 158

WMP Wireless Measurement Project. 60, 93, 94, 97, 103, 104

WPE Weighted Kriging Variance. 258–260, 263, 265, 267, 273, 277, 285, 295, 386