

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

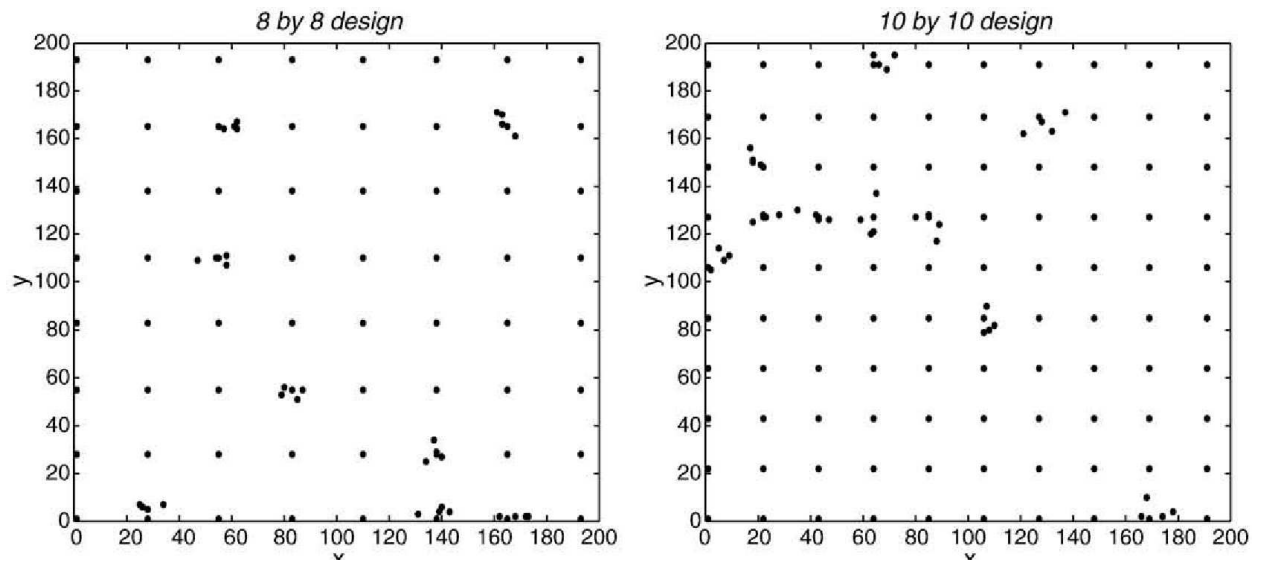


Figure 8.1: Systematic (grid) sampling augmented with nested "random" samples. In the $N \times N$ case, points are placed on an equally spaced regular grid of $N \times N$ points and then N additional points are sampled in the immediate neighborhood of N grid points. This has the effect of creating an initial sample that both covers the region and has a range of lag distances between measurements for variogram estimation. Figure taken from [63].

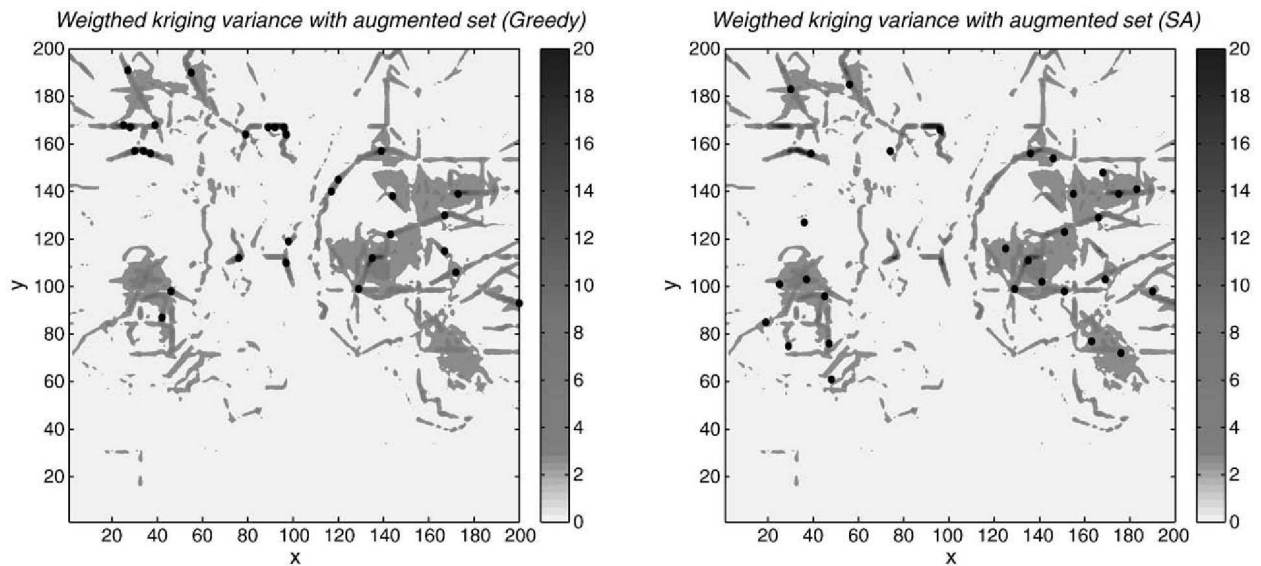


Figure 8.2: Optimized placement of second-phase samples according to the weighted Kriging variance proposed by Delmelle. Figure taken from [63].

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 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 achieve 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 “unmeasurable” 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 (ESRI) 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].

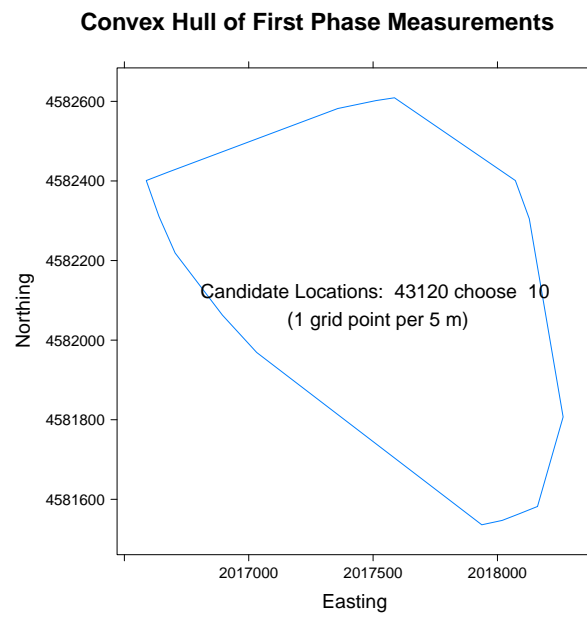
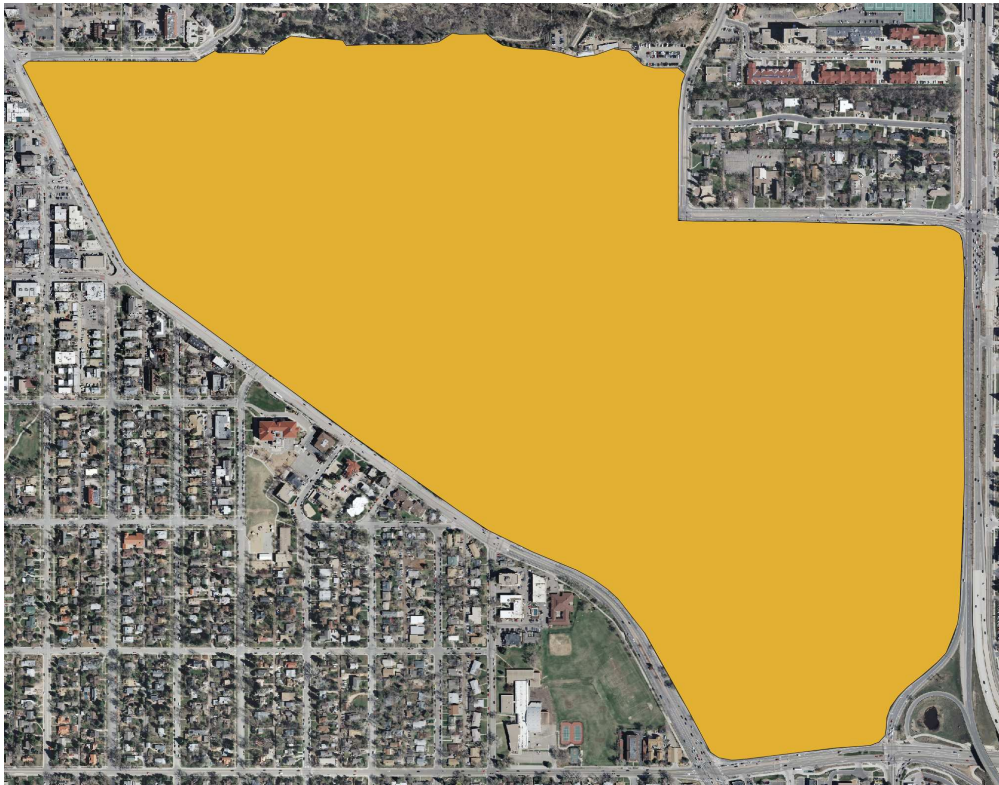
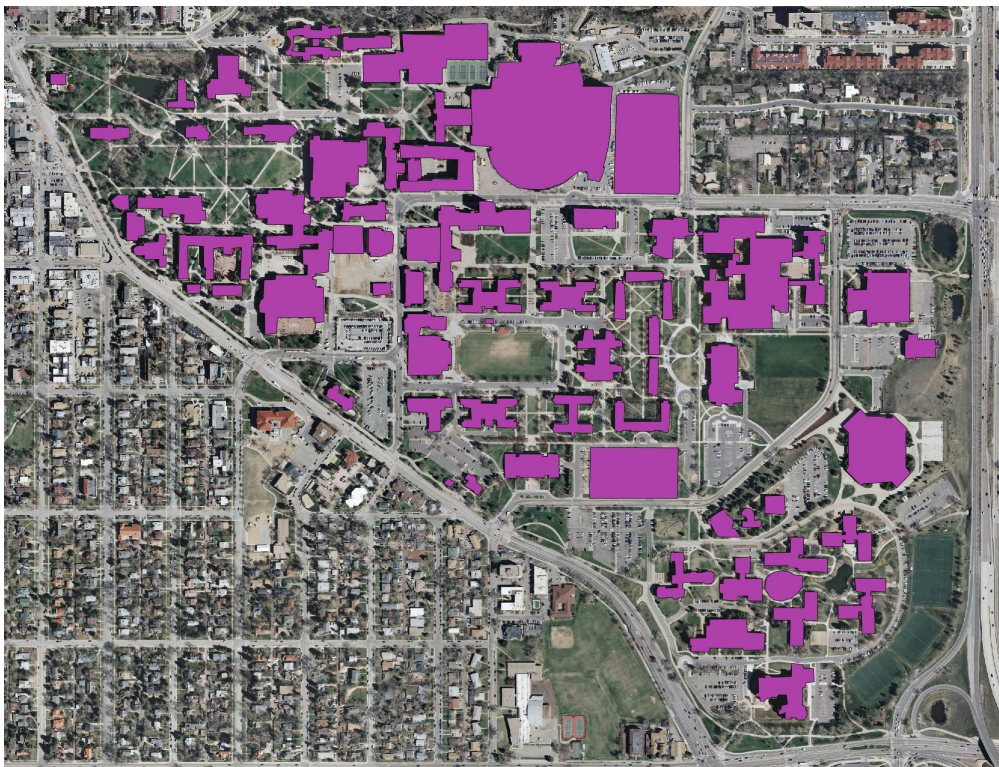


Figure 8.3: Convex hull around first-phase sample points for cuEN node.



(a) Measurement Boundary



(b) Unmeasurable Locations

Figure 8.4: Manual specification of boundary and unmeasurable 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]:

$$\sigma_E(\mathbf{x})^2 = \sigma^2 - k(\mathbf{x})^T K^{-1} k(\mathbf{x}) \quad (8.1)$$

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 $k(\mathbf{x})$ 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:

$$AKV = \frac{1}{N} \sum_i^N \sigma_E(\mathbf{x}_i) \quad (8.2)$$

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.

$$r(\mathbf{x}) = \sum_{j \in J} \frac{d(\mathbf{x}_j, \mathbf{x})^{-\beta} (Z(\mathbf{x}_j) - Z(\mathbf{x}))^2}{\sum_{k \in J} d(\mathbf{x}, \mathbf{x}_k)} \quad (8.3)$$

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:

$$WPE = \frac{1}{N} \sum_i^N \left(\frac{r(\mathbf{x}_i)}{R} \right)^\alpha \sigma_E(\mathbf{x})^2 \quad (8.4)$$

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: $n = 10, 20, \dots, 1000$. 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 “unmeasurable” 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

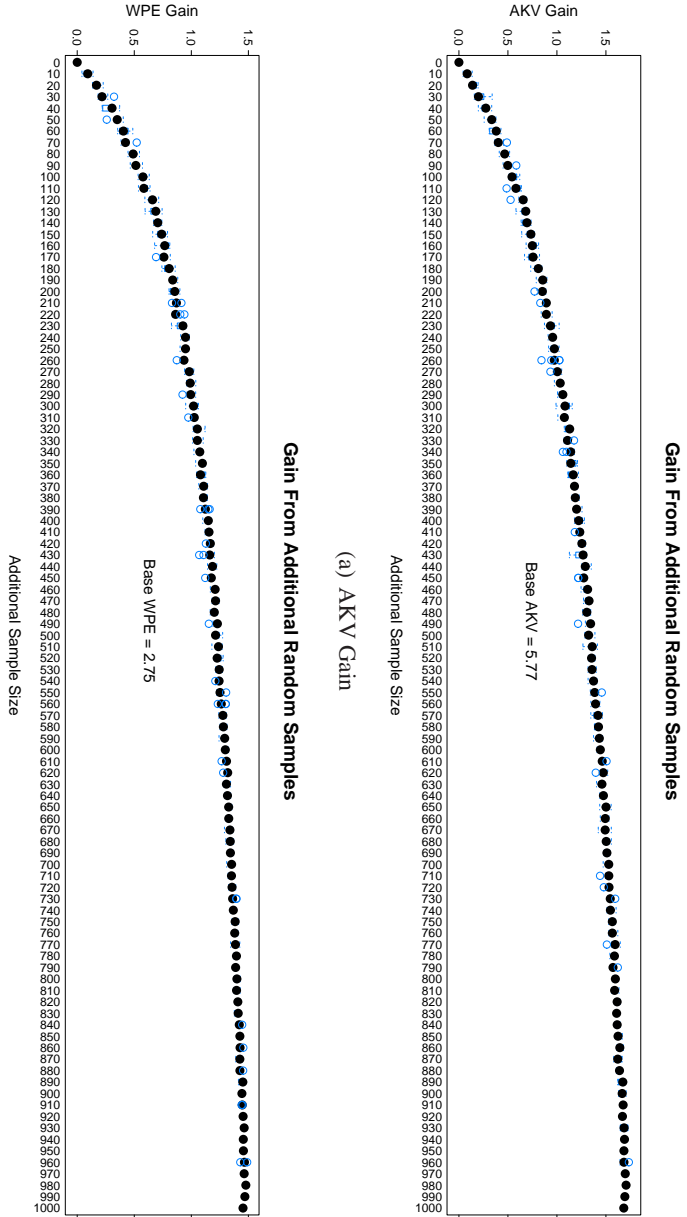


Figure 8.5: Performance gain from increasingly large simple random samples.

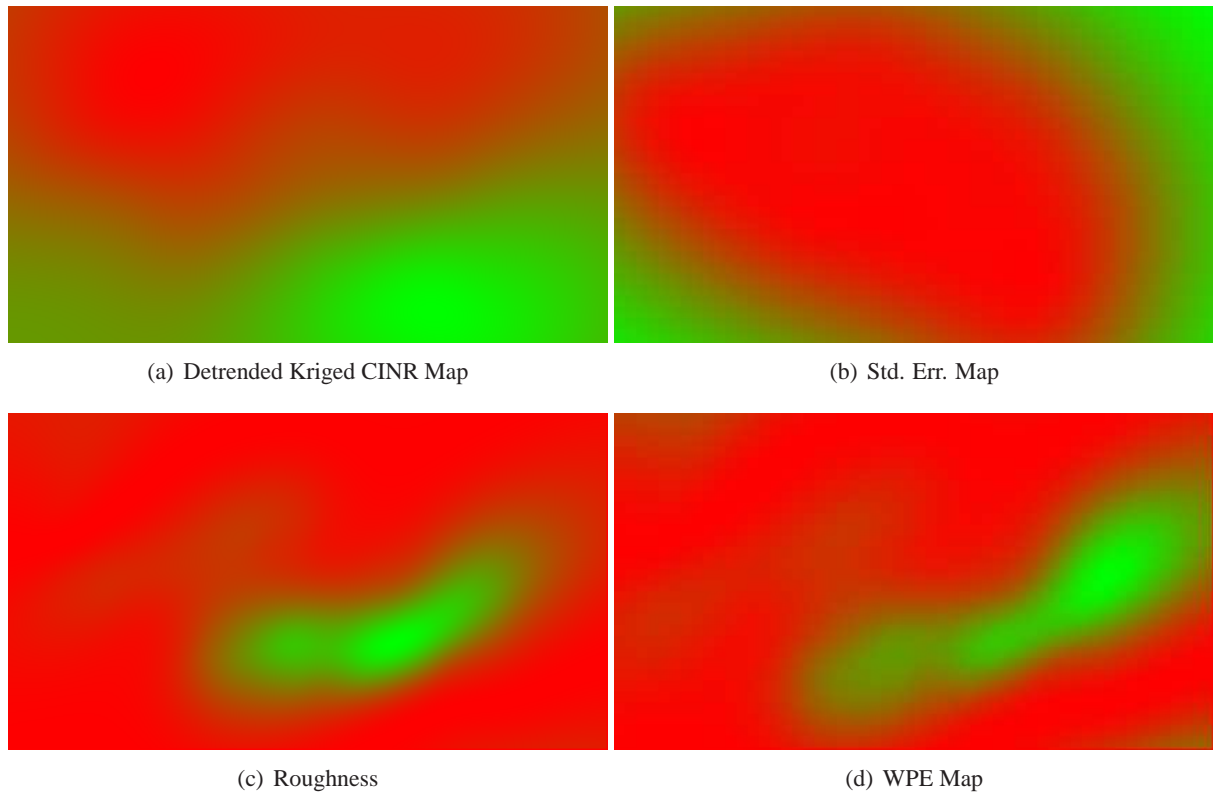


Figure 8.6: Maps used for sample optimization for cuEN node.

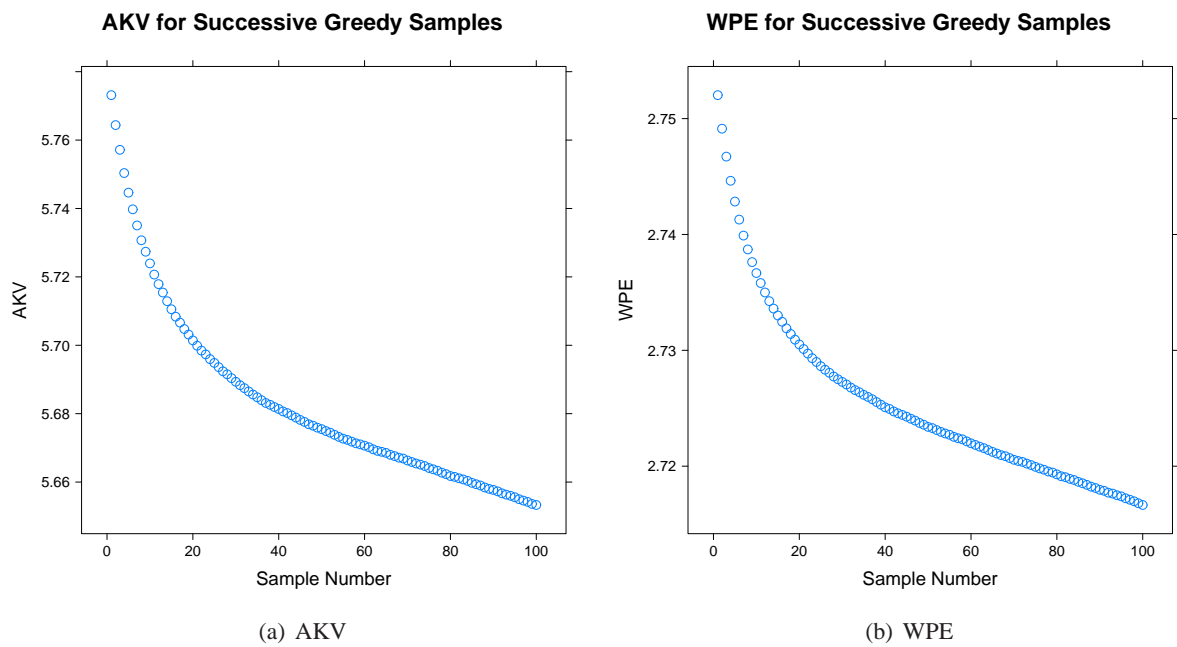


Figure 8.7: Improvement (gain) from iterative greedy sampling using the WPE map.

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 constraints 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:

$$Pr[accept] = e^{\frac{\Delta f}{t}} \quad (8.5)$$

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

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 achieved. 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

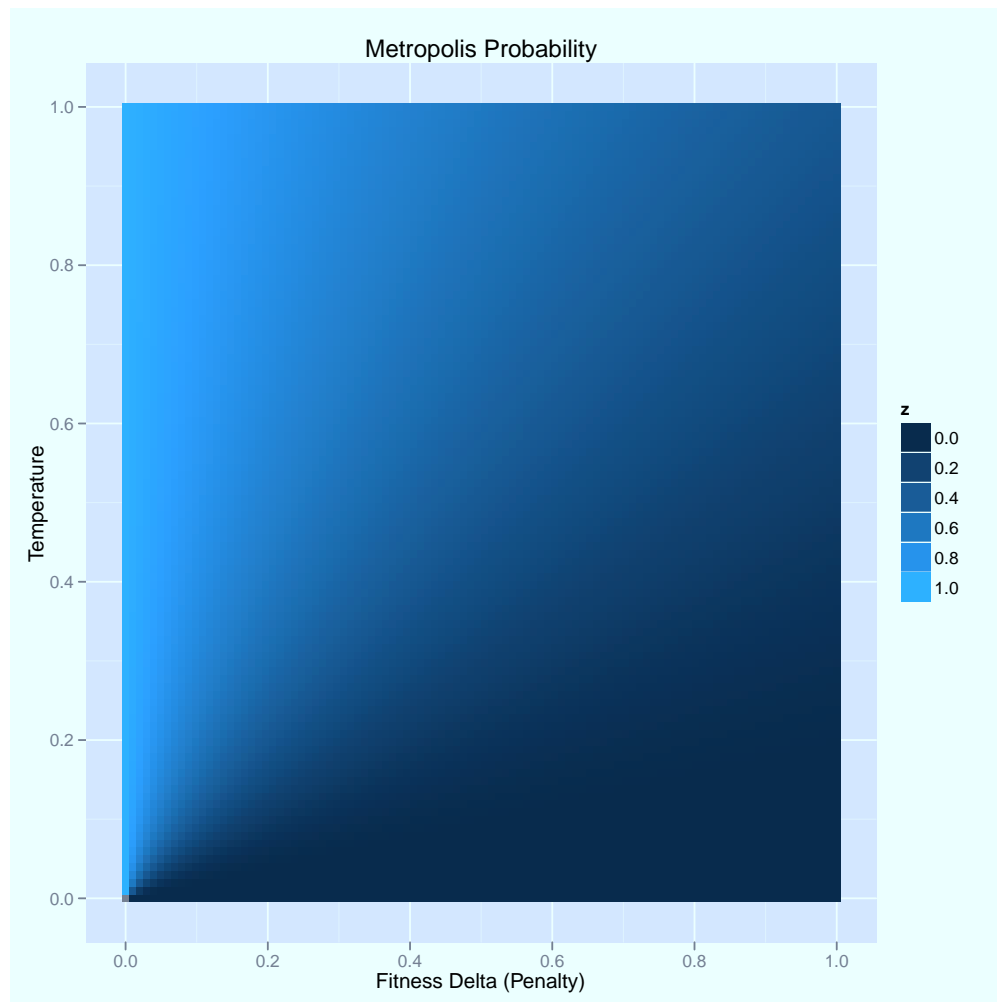


Figure 8.9: Metropolis probability plot for spatial simulated annealing.

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.

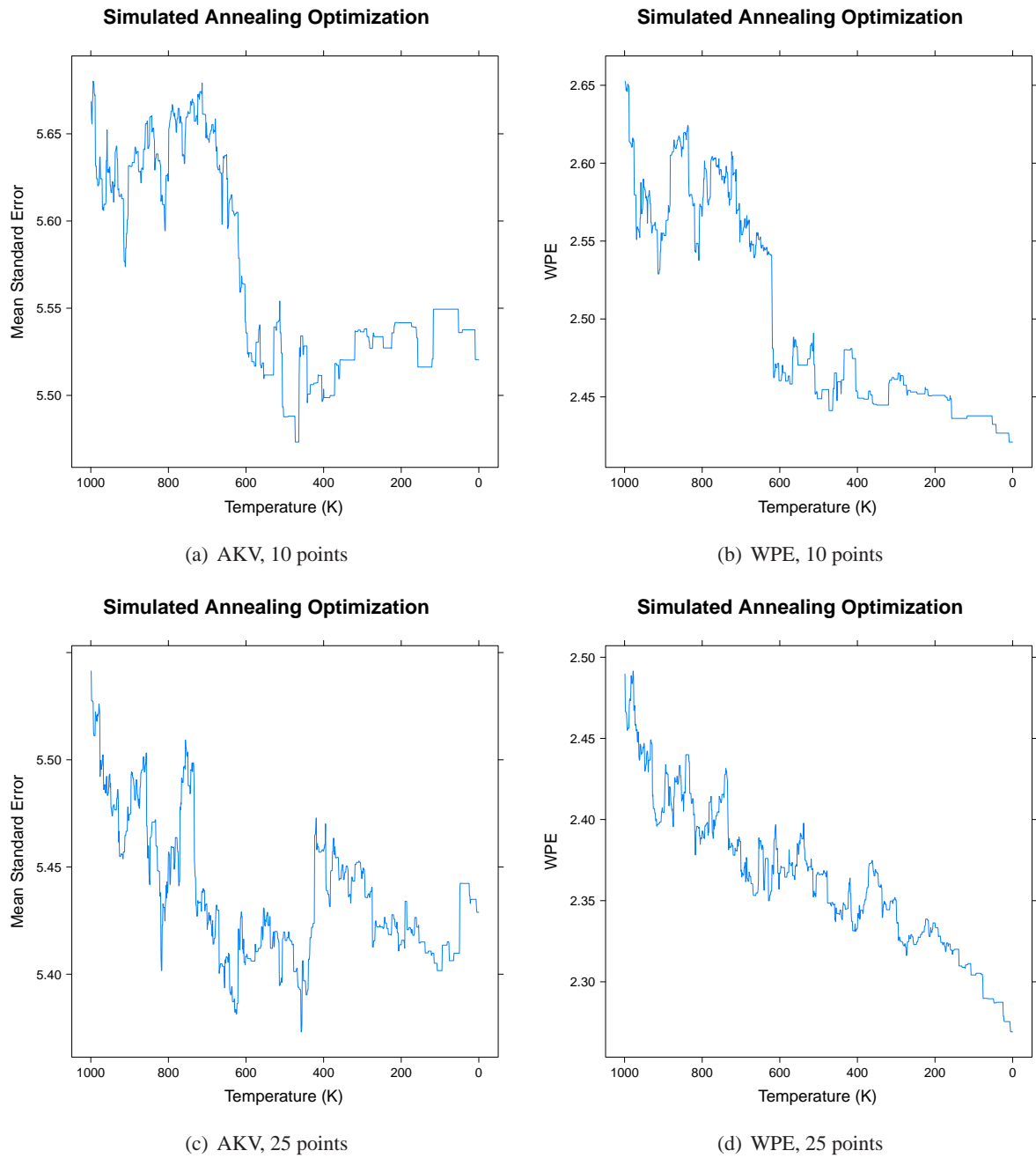


Figure 8.10: Improvement (gain) from spatial simulated annealing using the WPE map for sets of 10 or 25 additional points.

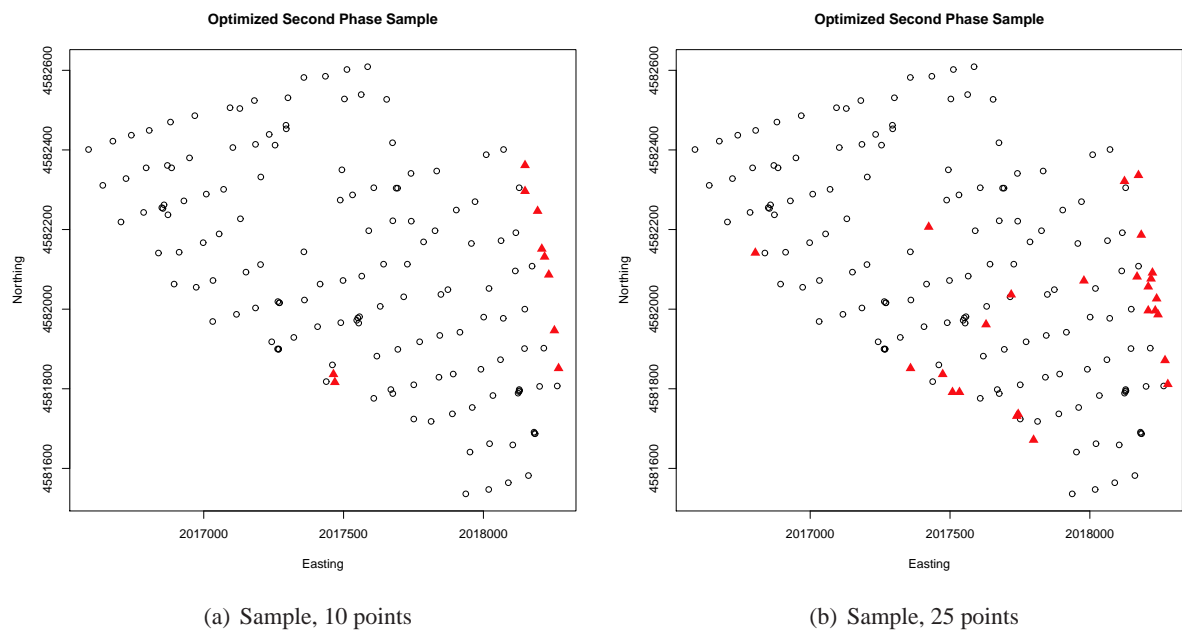


Figure 8.11: Optimized sample for 10 and 25 points after 1000 iterations.

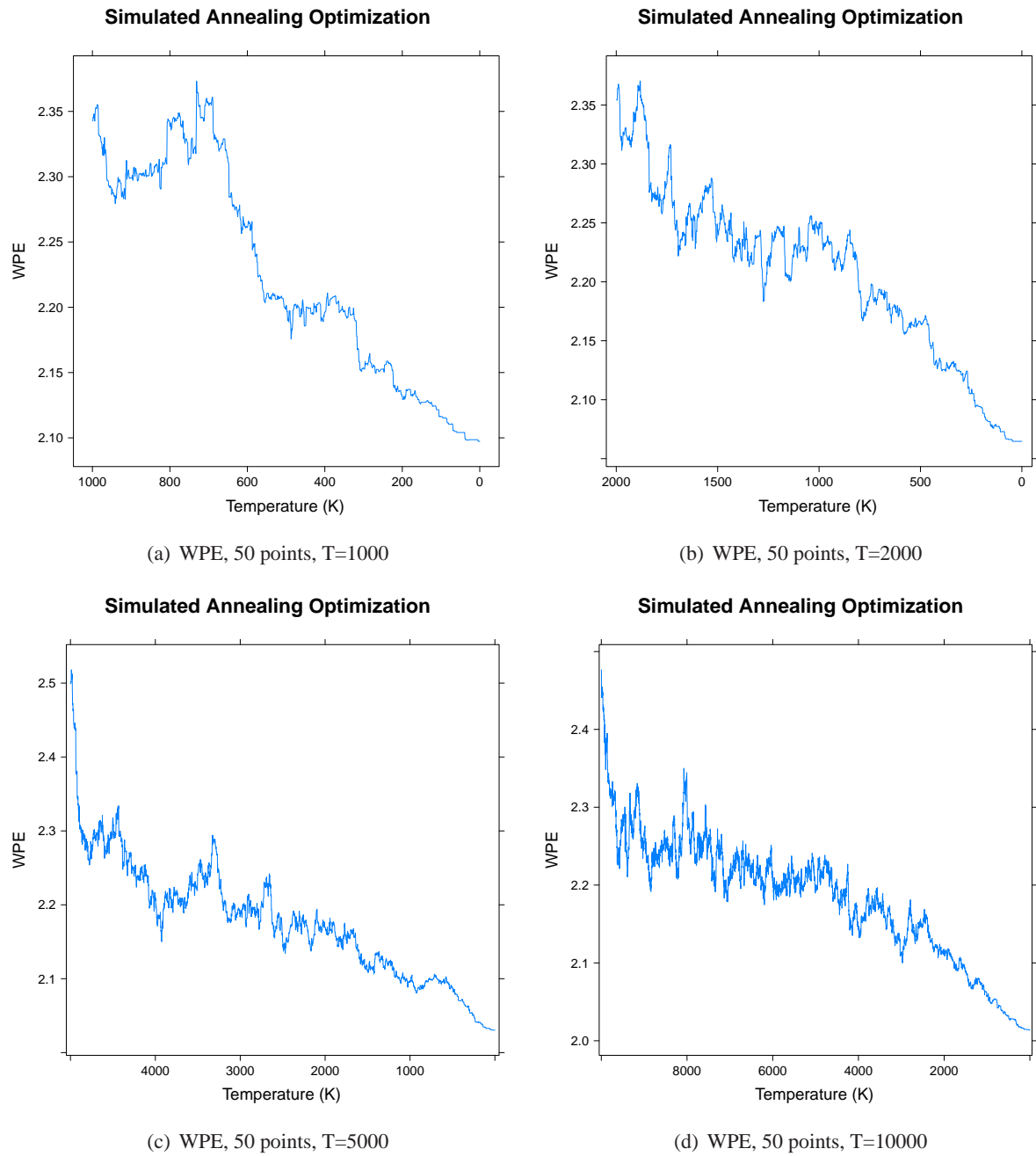


Figure 8.12: Improvement (gain) from spatial simulated annealing using the WPE map for 50 additional points with increasing numbers of iterations

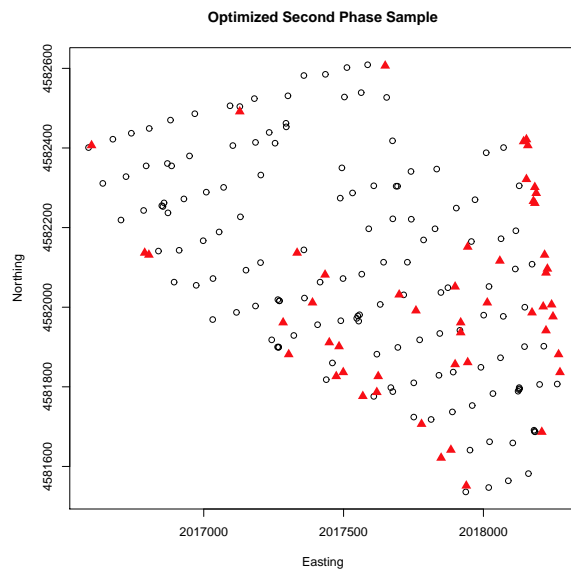
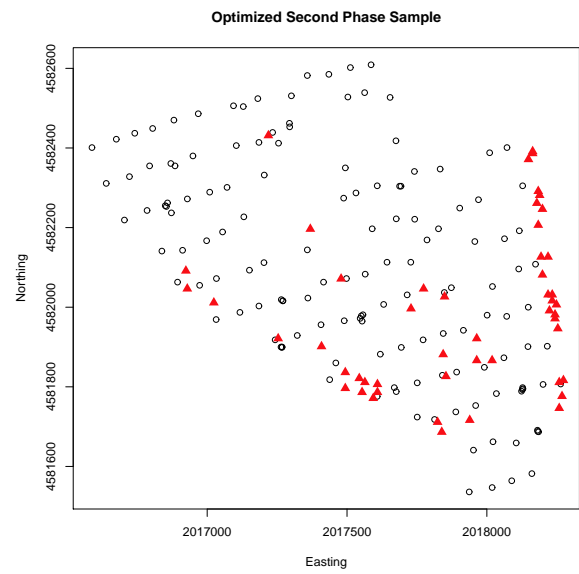
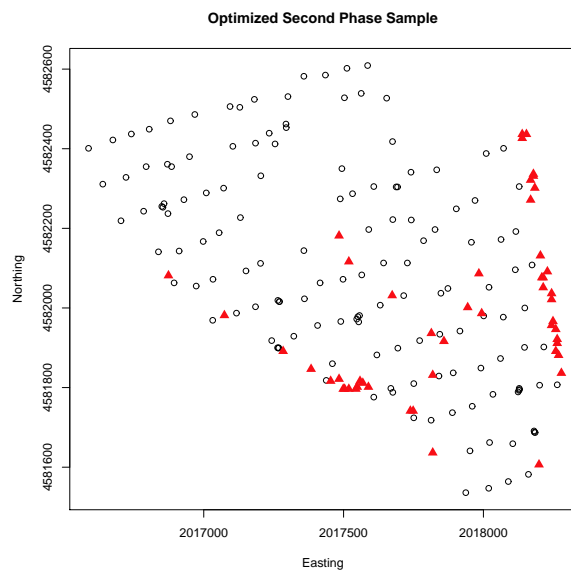
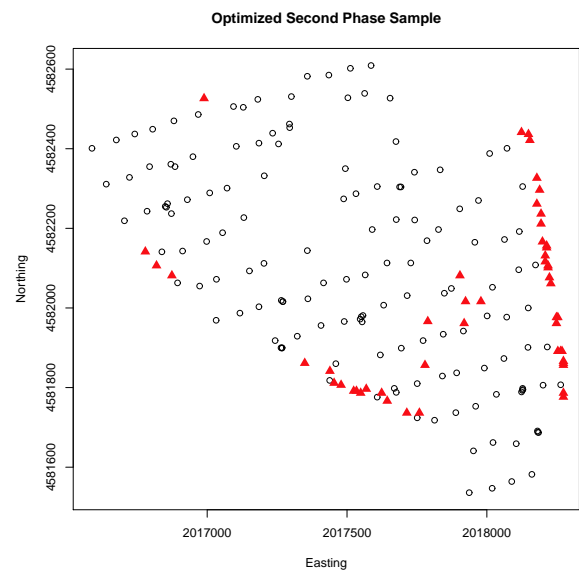
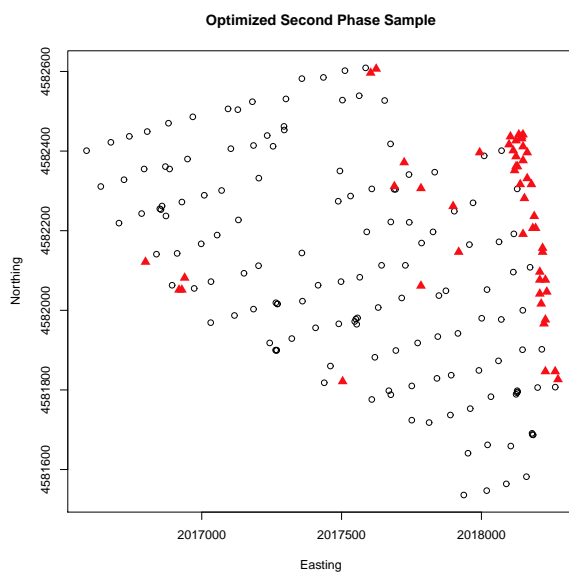
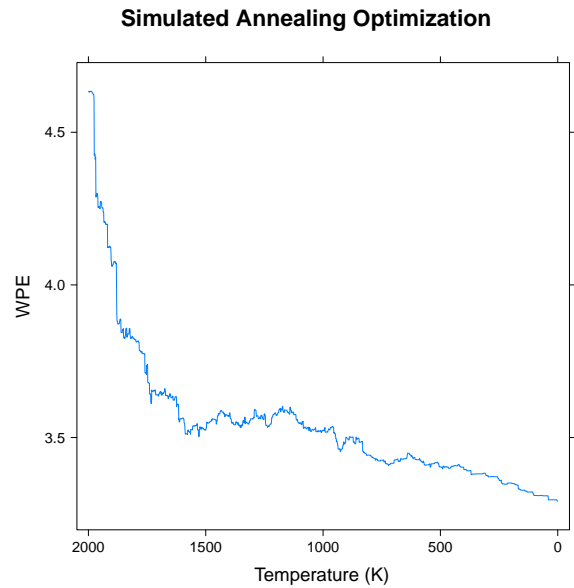
(a) WPE, 50 points, $T=1000$ (b) WPE, 50 points, $T=2000$ (c) WPE, 50 points, $T=5000$ (d) WPE, 50 points, $T=10000$

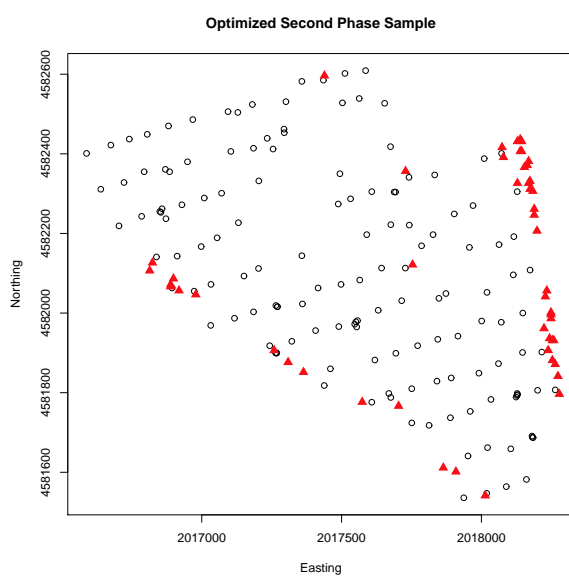
Figure 8.13: Optimized sample using the WPE map for 50 additional points with increasing numbers of iterations



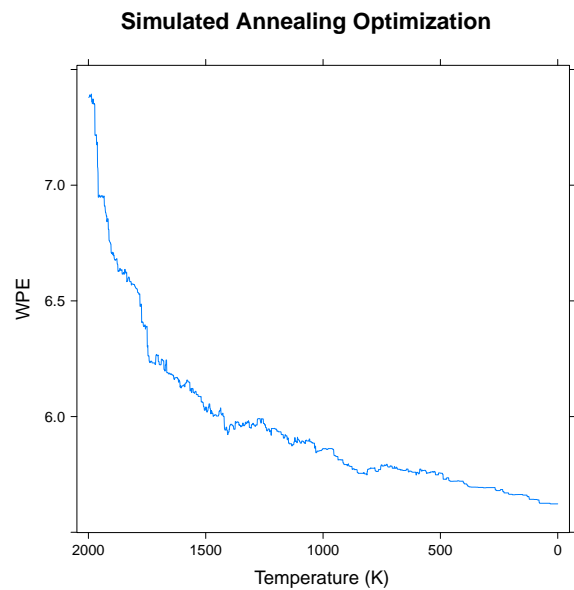
(a) ESNR6, 50 points, T=2000



(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 ineffective 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 bimodal. 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,

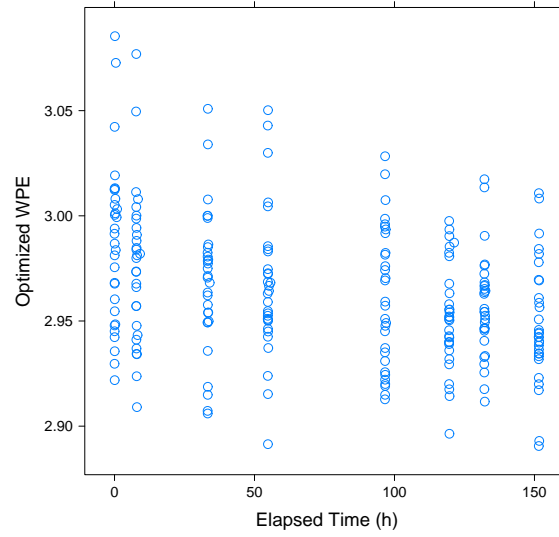
which can be seen centered around zero. After about 100 hours, or 150 slave runs (30 per run, 5 queue runs) all of the low-hanging fruit have been picked from the pool, and the remaining gain to be extracted is marginal.

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.

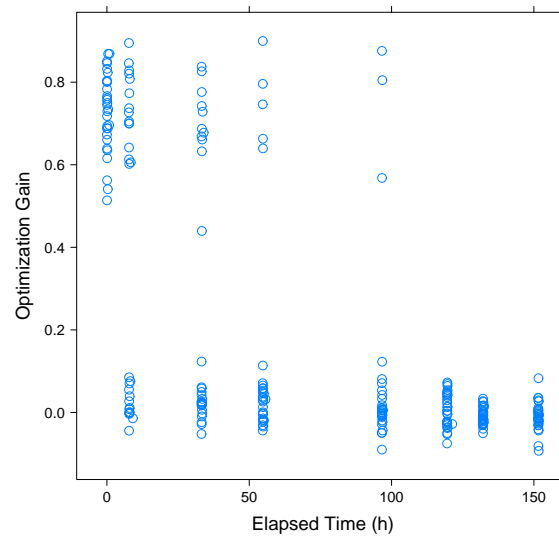
For these experiments, the Janus supercomputer is used, which has 1368 compute nodes total, each of which has 12 Central Processing Unit (CPU) cores and 2 GB of Random Access Memory (RAM) per core [156]. The OpenMPI message passing library and a master/slave architecture are used [220]. The master is 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.

Parallel Metaheuristic Optimization, N=25



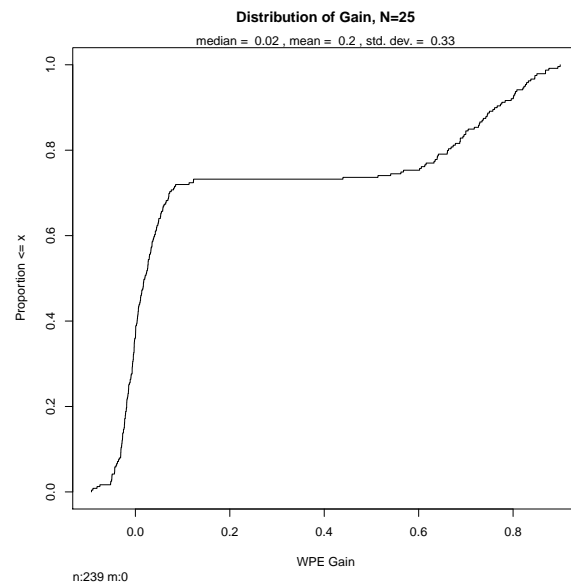
(a) WPE

Parallel Metaheuristic Optimization, N=25



(b) WPE Gain

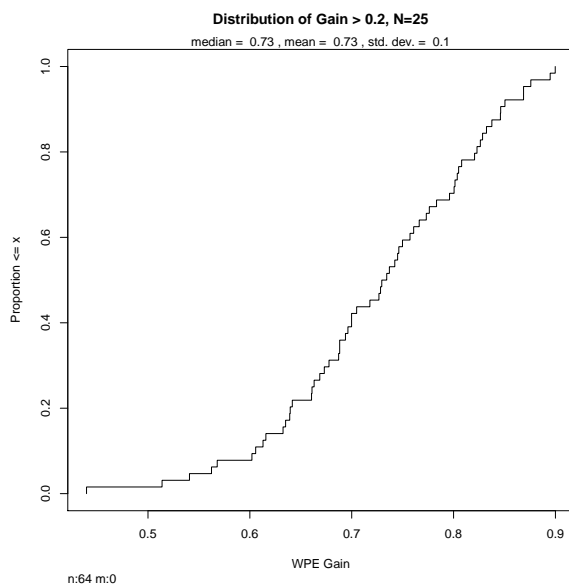
Figure 8.15: Optimization timeseries for $N = 25$.



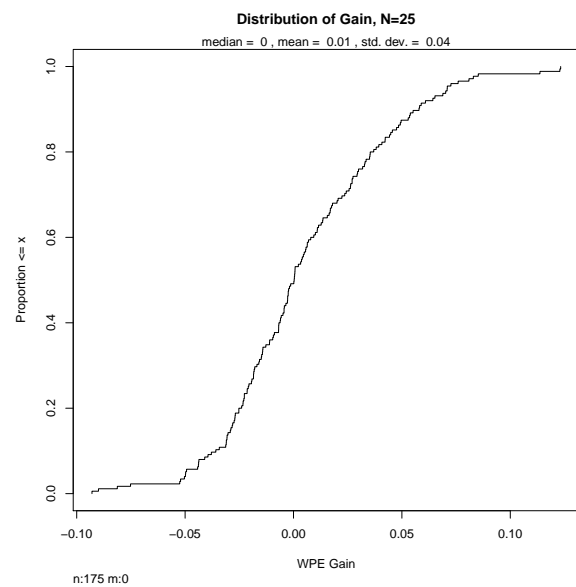
(a) Complete Distribution



(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 achieves 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 achieves 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 achieve 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 $T = 2000$ achieves 2.04 and the best with $T = 1000$

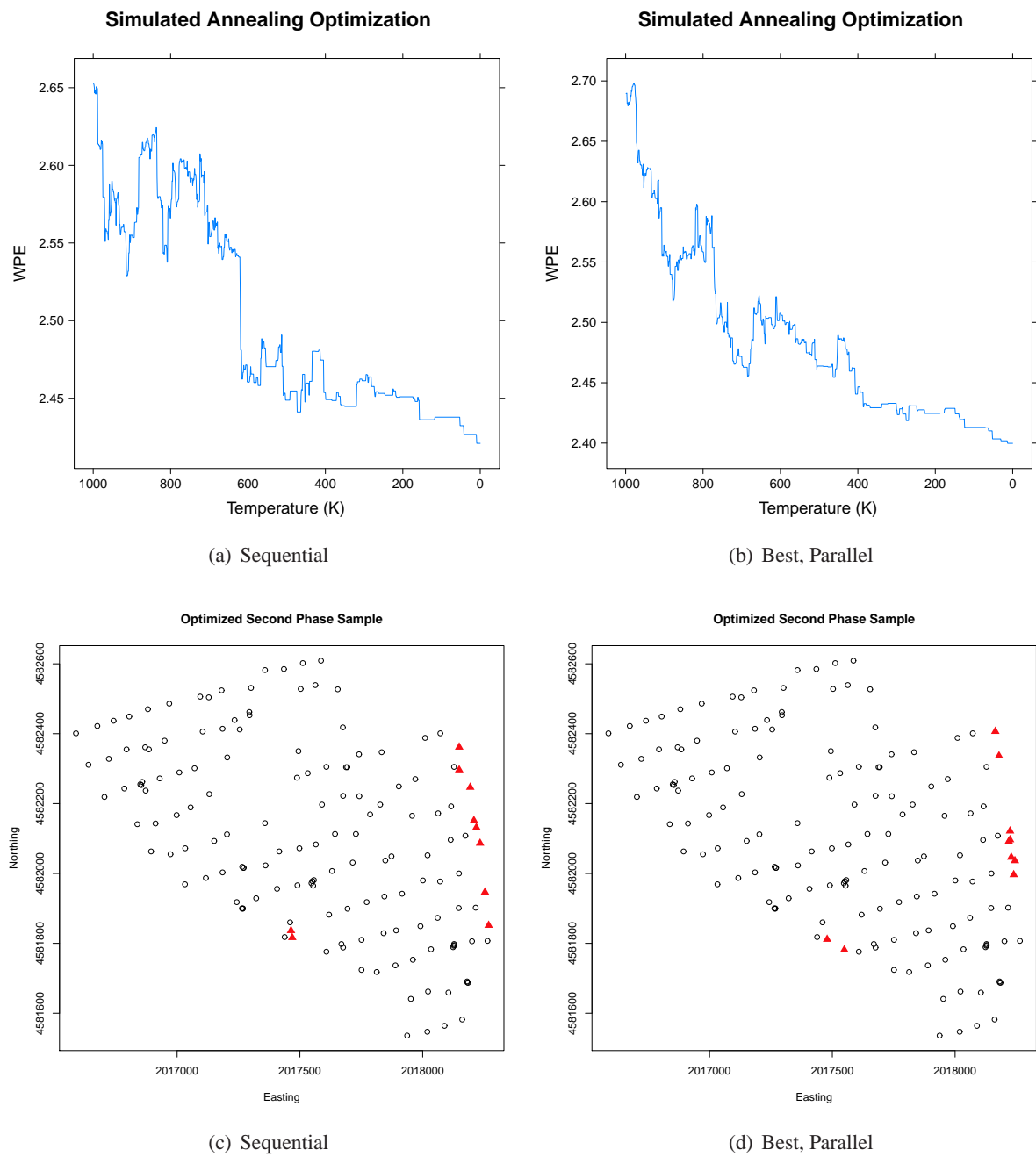


Figure 8.17: Parallel versus sequential performance for $N = 10$.

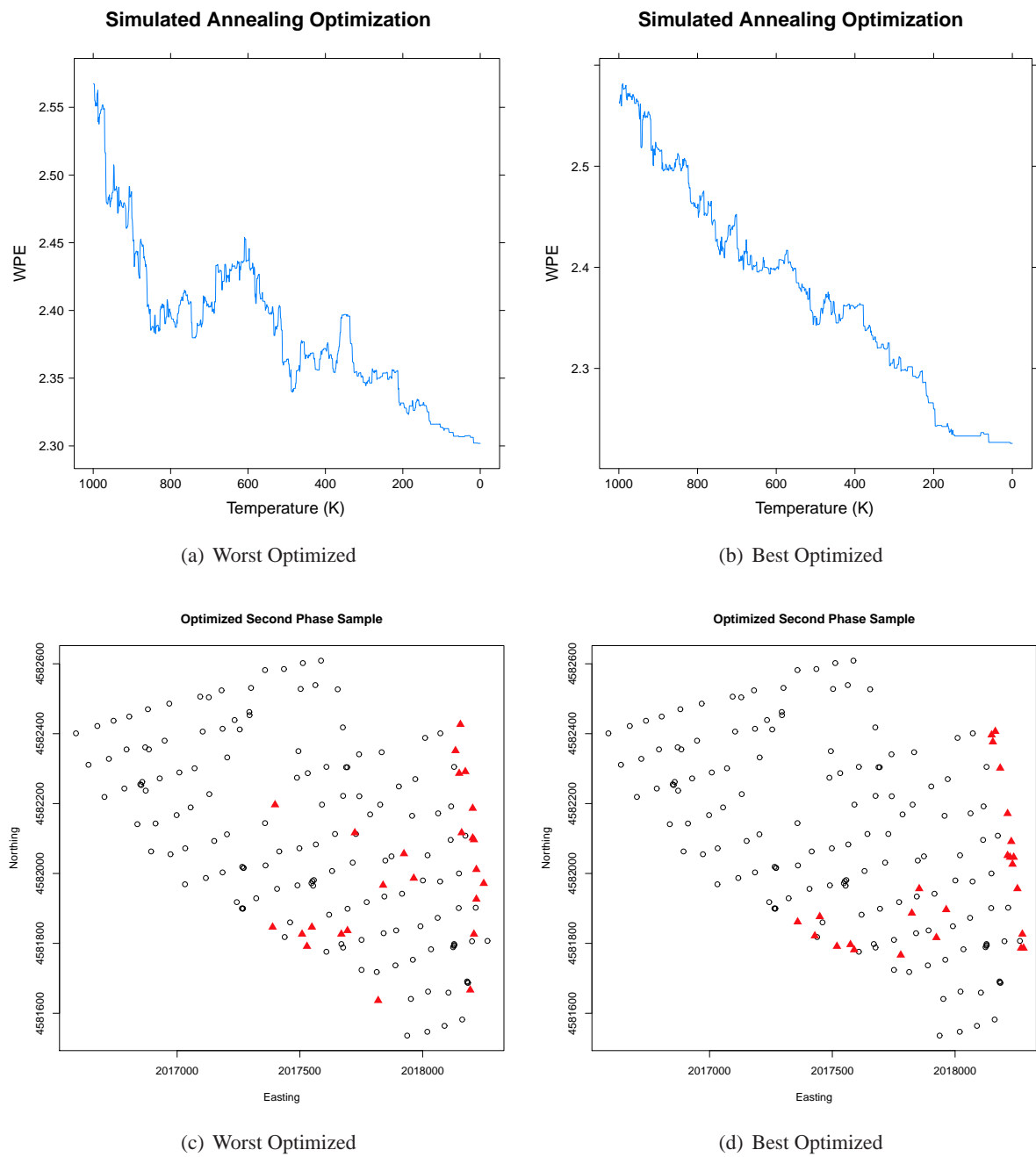
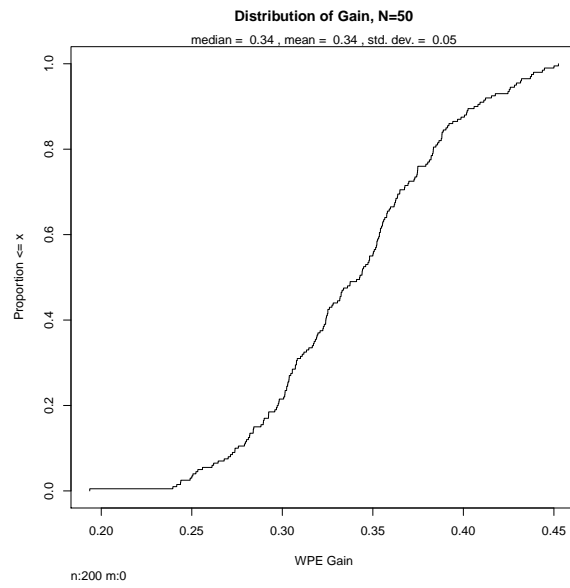
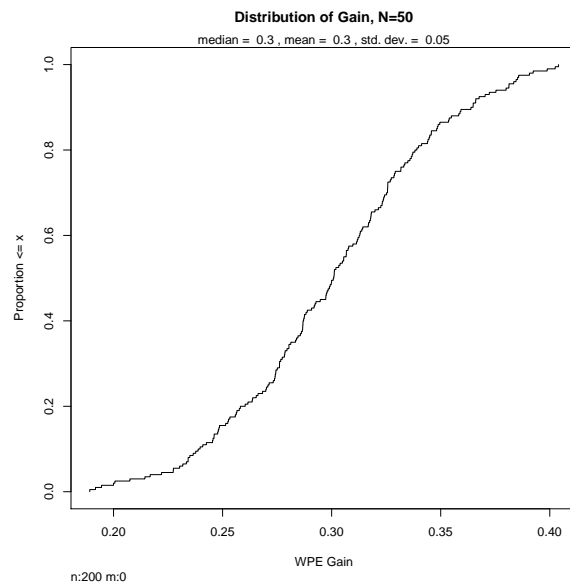


Figure 8.18: Parallel versus sequential performance for $N = 25$.

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 achieved. 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 thoroughness) 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 optimizations 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.

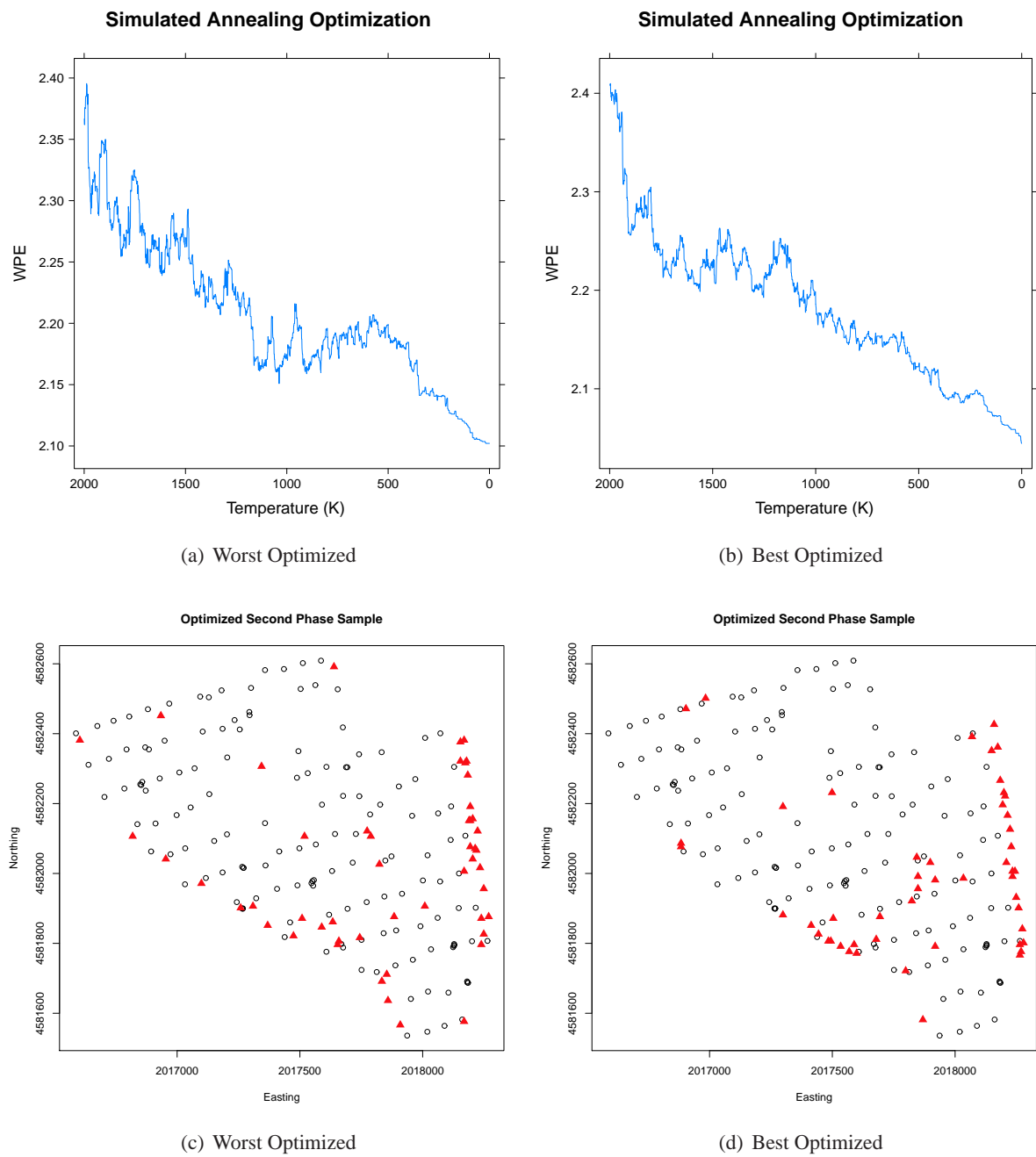


Figure 8.20: Parallel versus sequential performance for $N = 50$.

$N = 50$ sample overlayed on Google Maps orthoimagery. The measurement procedure 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 simultaneous 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 unmeasurable 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 procedure 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

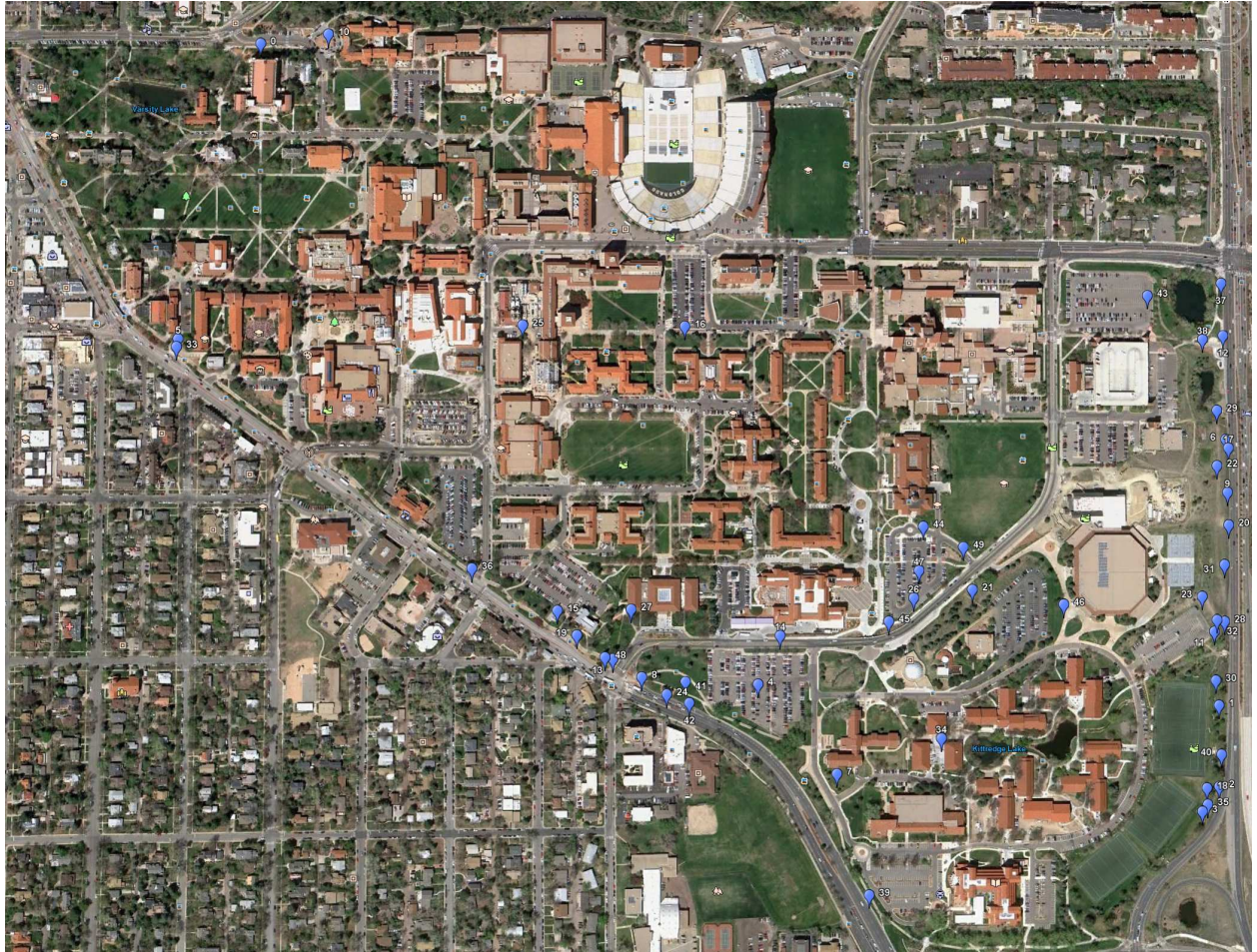


Figure 8.21: Optimized second-phase sample for cuEN node with 50 points, overlaid on Google Maps orthoimagery.

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.

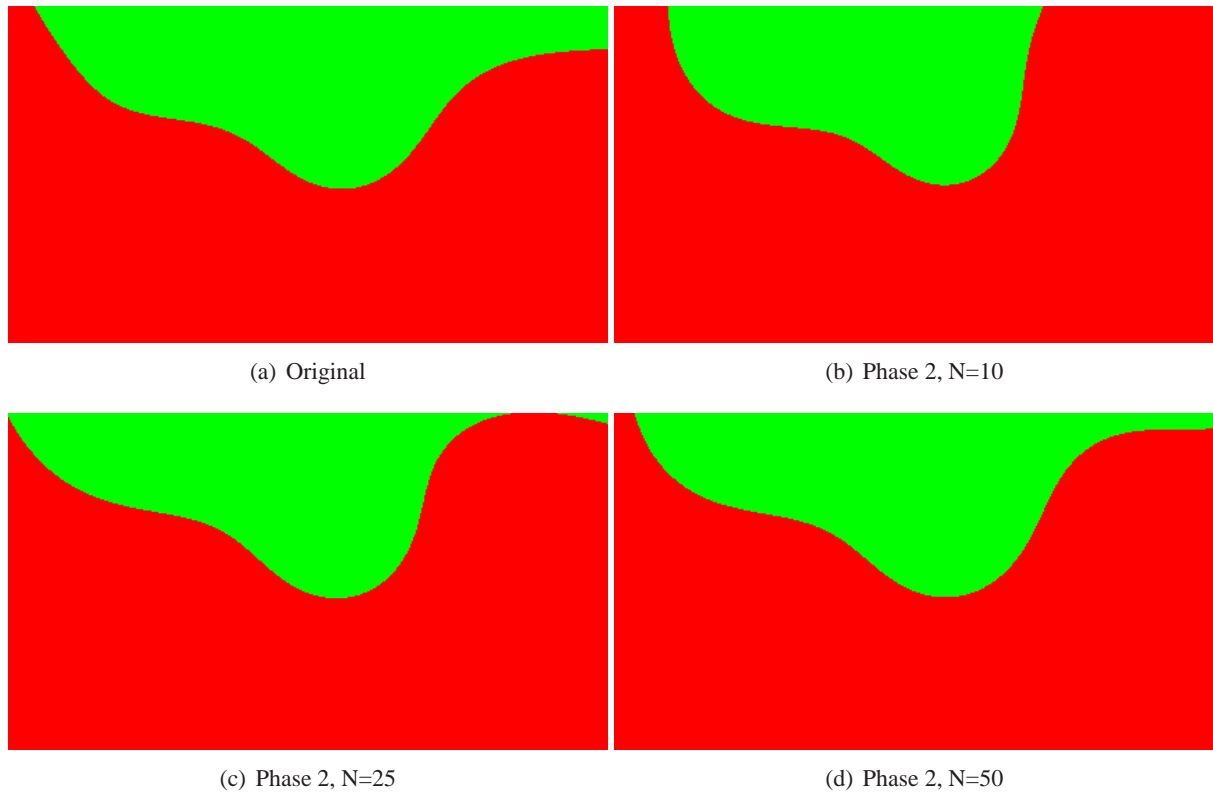


Figure 8.22: Comparison of second-phase threshold maps for CINR=20.

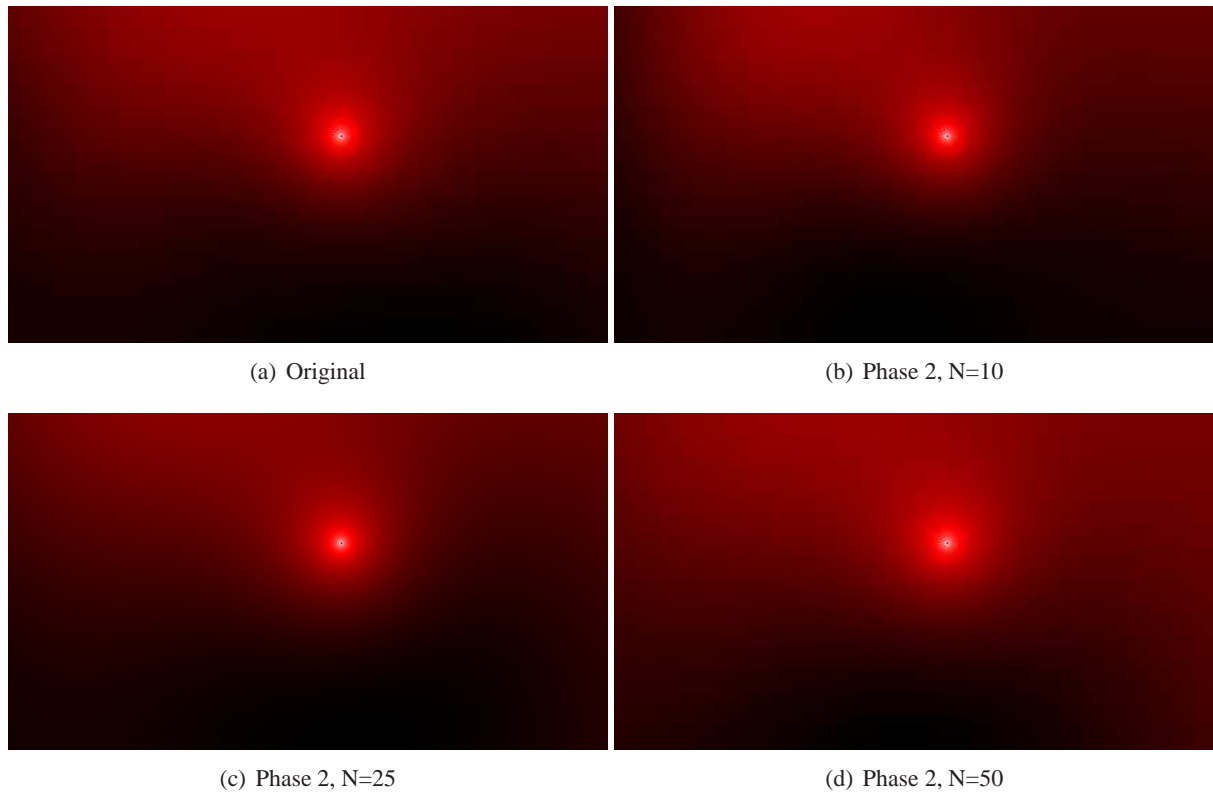


Figure 8.23: Comparison of second-phase coverage.

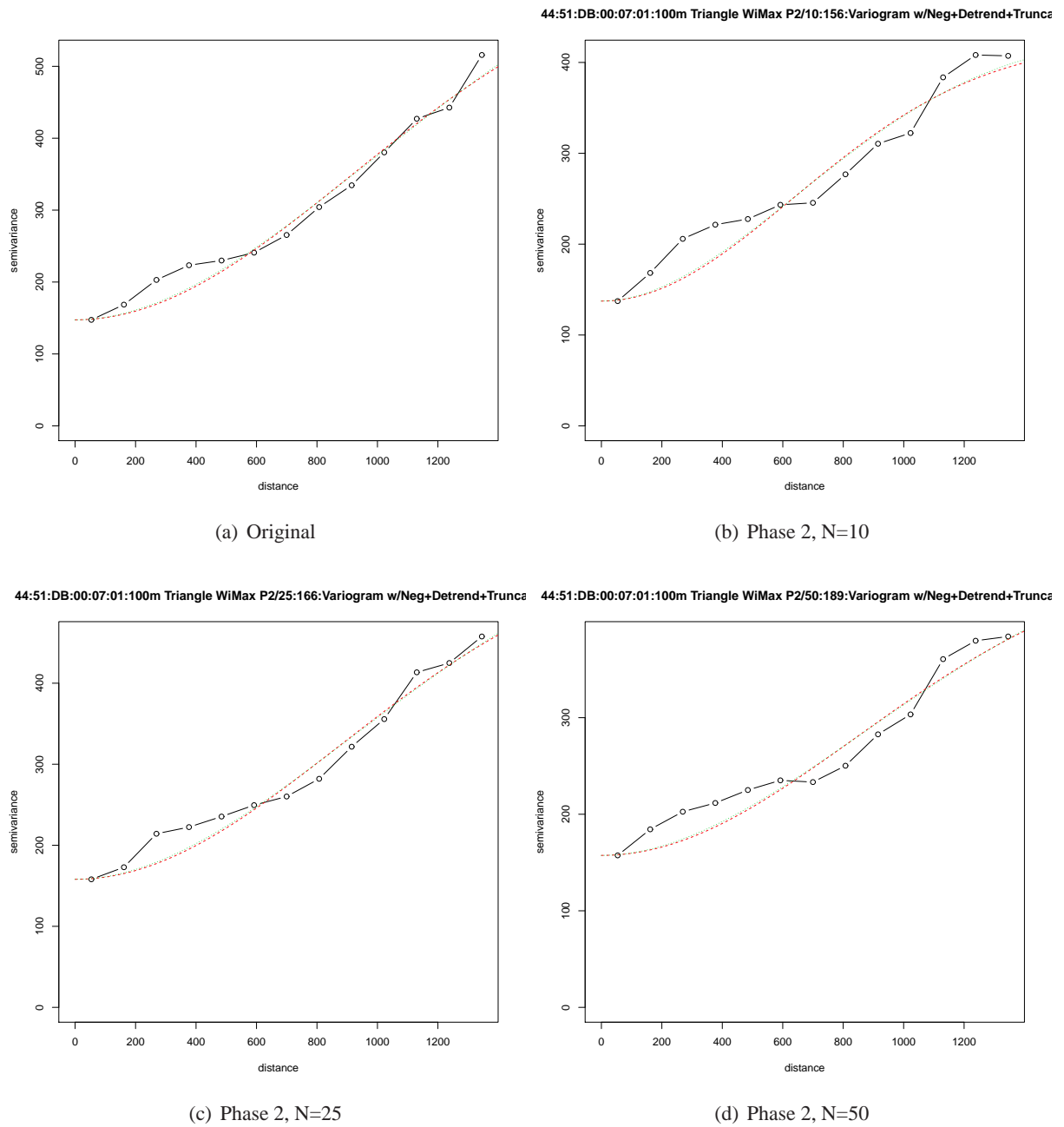


Figure 8.24: Comparison of second-phase variograms.

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%

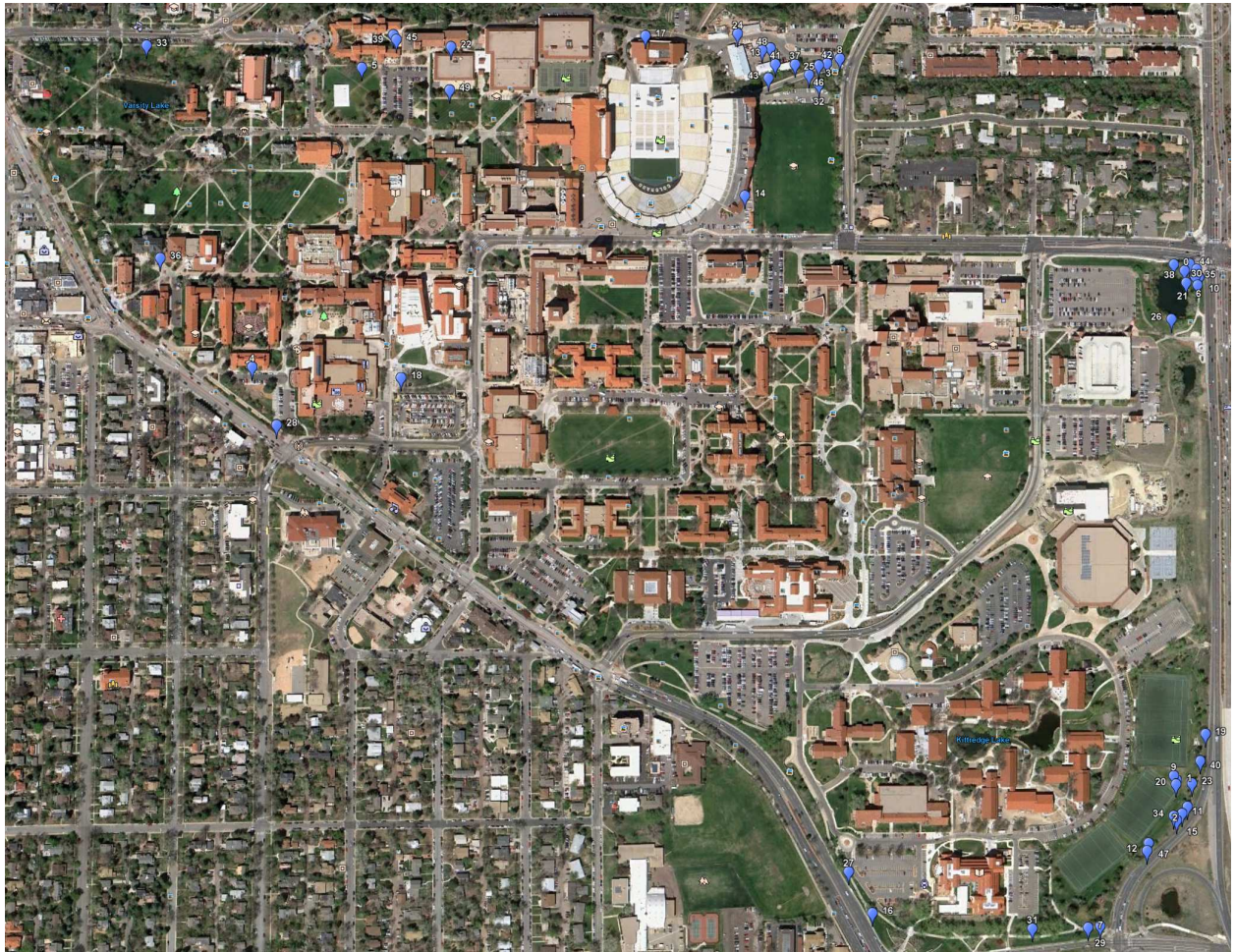


Figure 8.25: Measured second-phase sample for cuEN node with 50 points, overlaid on Google Maps orthoimagery.

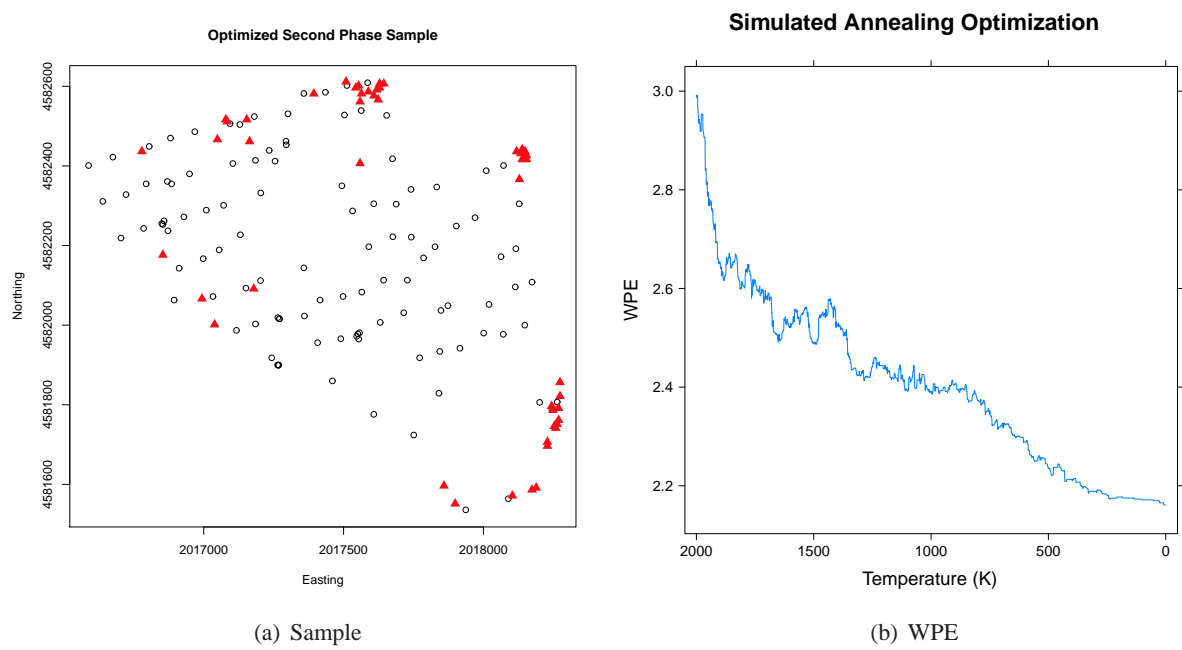


Figure 8.26: Phase-2 Optimized Sample.

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 unmeasurable 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.

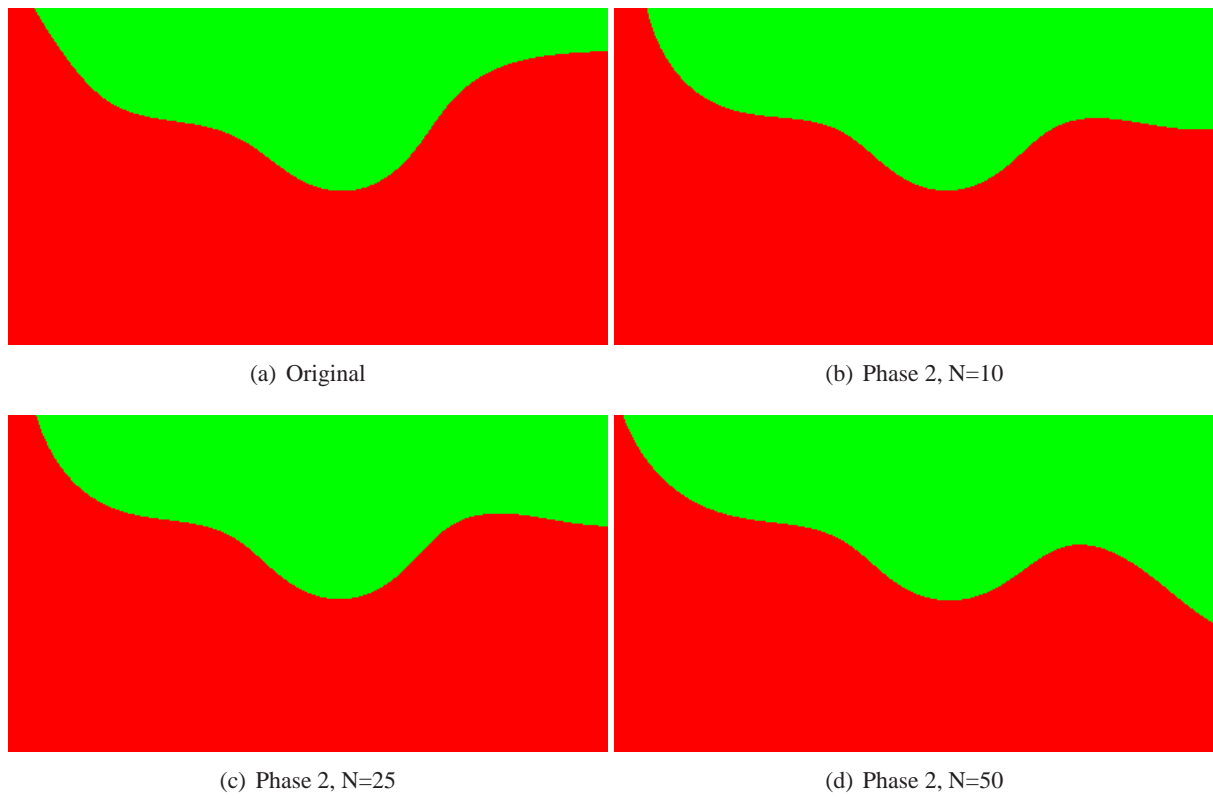


Figure 8.27: Comparison of second-phase threshold maps (threshold is CINR = 40 dB).

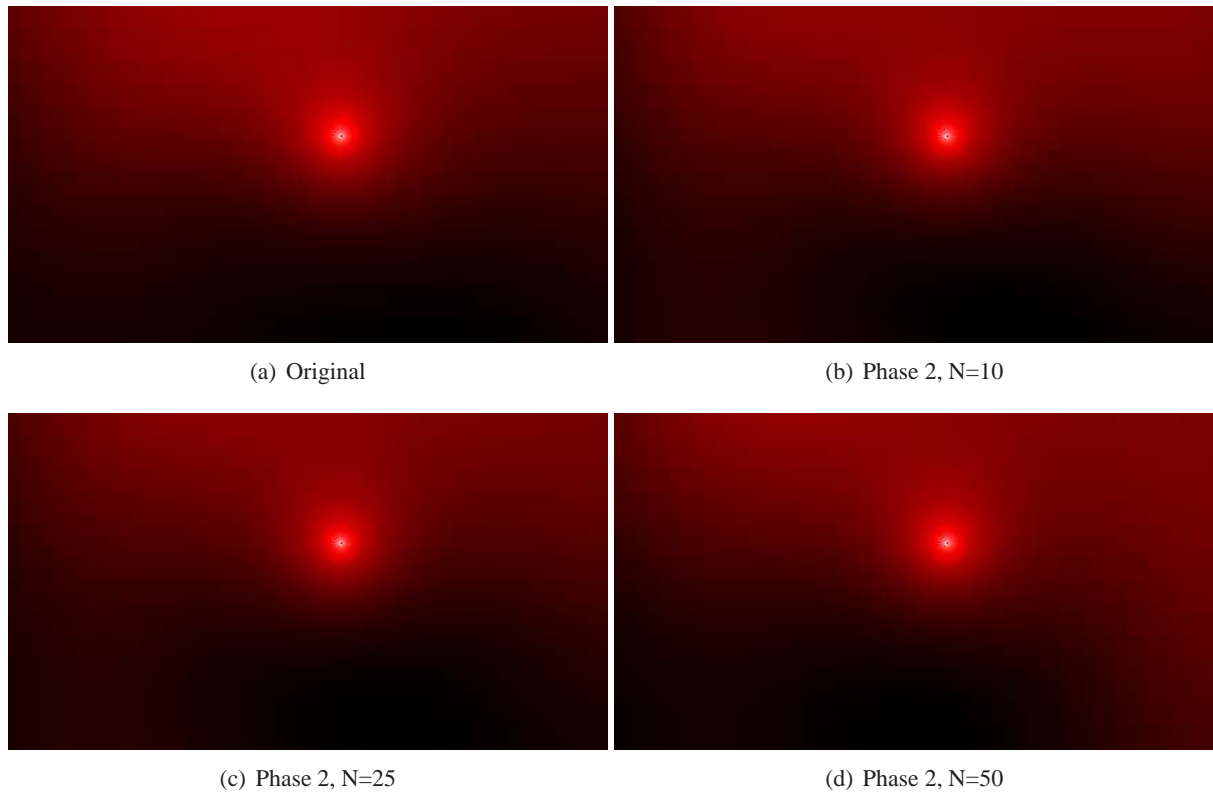


Figure 8.28: Comparison of second-phase coverage.

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 prominent 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-

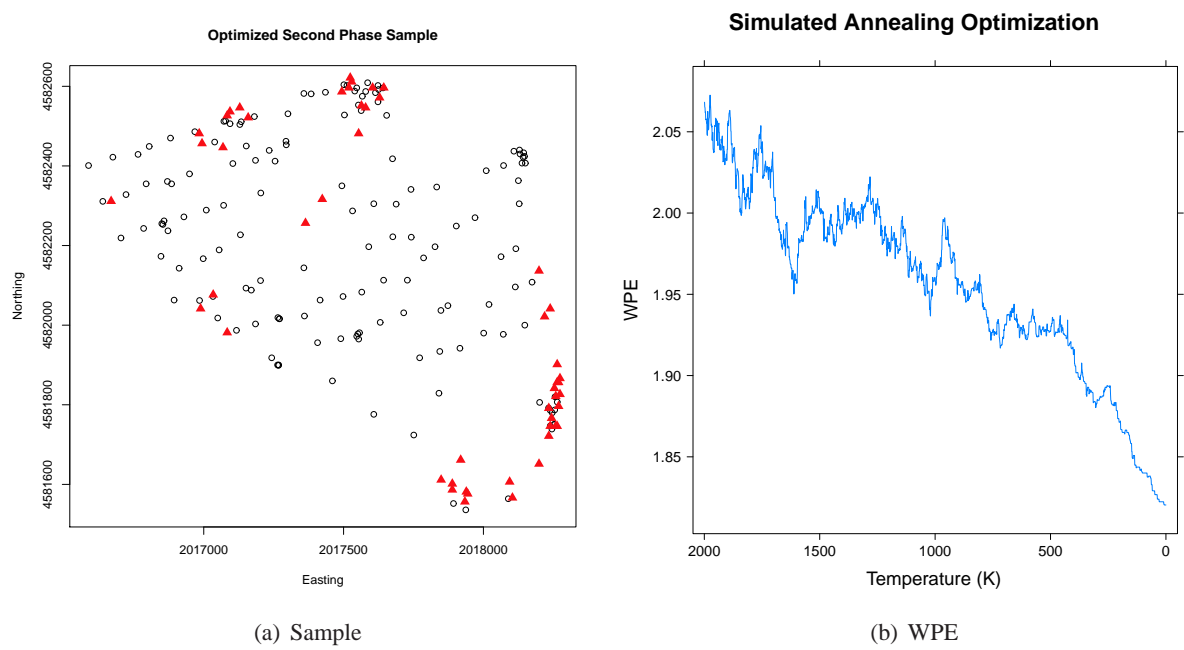


Figure 8.29: Phase-3 Optimized Sample.

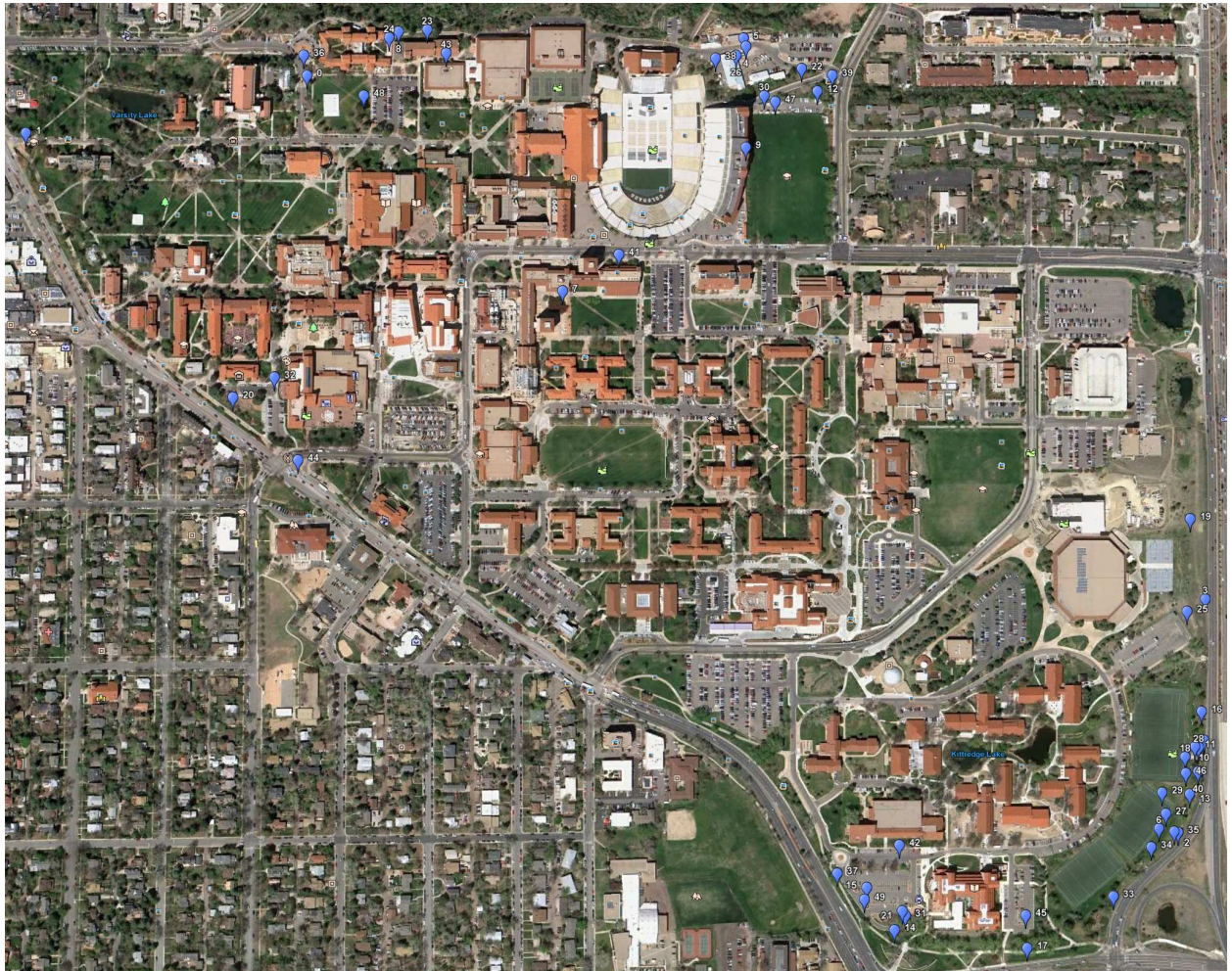


Figure 8.30: Measured third-phase sample for cuEN node with 50 points, overlaid on Google Maps orthoimagery.

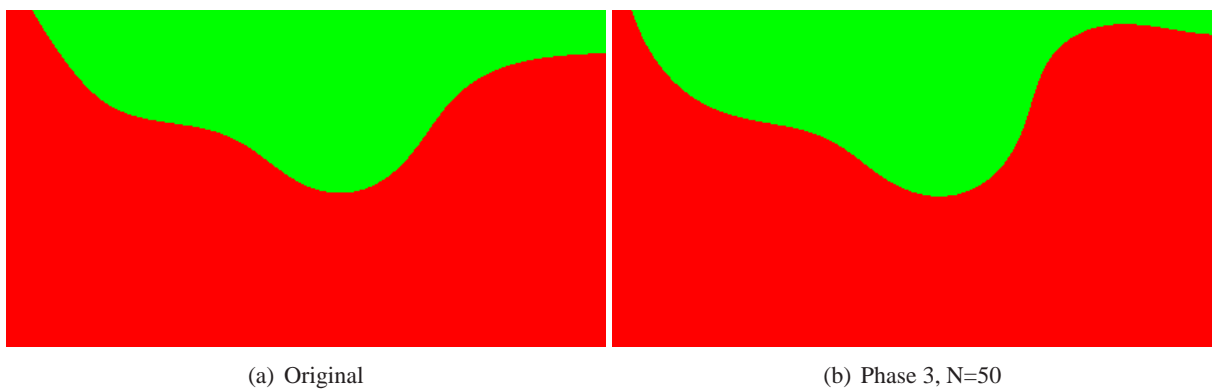


Figure 8.31: Comparison of second-phase threshold maps (threshold is CINR = 40 dB).

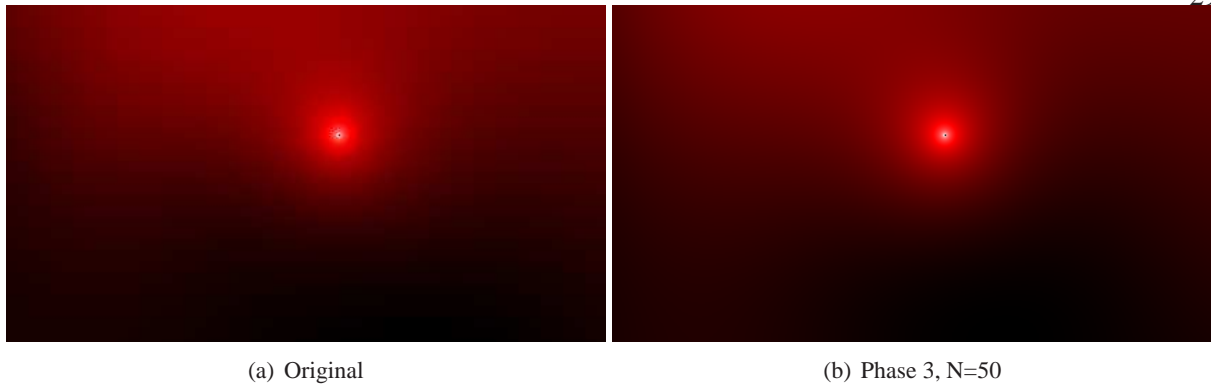


Figure 8.32: Comparison of second-phase coverage.

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 unnecessary is an important area for future investigation.