

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**AUTONOMOUS FIELD EXPLORATION VIA KRIGING
PREDICTION VARIANCE SUPPRESSION**

A thesis submitted in partial satisfaction of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

with an emphasis in ROBOTICS AND CONTROL

by

Sargis S Yonan

December 2018

The Thesis of Sargis S Yonan
is approved:

Professor Gabriel Hugh Elkaim, Chair

Professor Renwick E. Curry

Professor Patrick E. Mantey

Lori Kletzer
Vice Provost and Dean of Graduate Studies

Copyright © by

Sargis S Yonan

2018

Table of Contents

List of Figures	vi
Abstract	1
Dedication	2
Acknowledgments	3
1 Introduction	4
1.1 Previous Works	6
1.2 Problem Definitions	10
1.2.1 The Field	10
1.2.2 The Sensor	11
2 Spatial Analysis	12
2.0.1 Autocorrelation in a Field	13
2.1 Inverse Distance Weighting	14
2.2 Variography	16
2.2.1 The Variogram	16
2.2.2 The Semivariogram	17
2.2.3 The Empirical Semivariogram	18
2.2.4 Converting a Semivariogram to a Variogram	18
2.2.5 Variogram Models	18
2.2.6 Fitting A Semi-Variogram	20
2.3 The Kriging Method	22
2.3.1 Forms of the Kriging Method	23
2.3.2 Covariance Matrix From A Variogram	23
2.3.3 The Proximity Vector	24
2.3.4 The Kriging Weights	25
2.3.5 The Kriging Prediction Equation	25

2.3.6	Variance of A Kriging Prediction	25
2.3.7	Procedure For Field Prediction Using The Kriging Method	26
3	Path Planning	29
3.1	Vehicle & Information Gain Model	29
3.1.1	Exploration Vehicle Model Dynamics	30
3.1.2	Field Uncertainty Model	30
3.1.3	Uncertainty Loss Function	31
3.2	Next Highest Variance Path Planner	31
3.2.1	Inefficiency in NHV	32
3.3	N Next Highest Variances Path Planner	32
3.4	Monte Carlo Path Planner	34
3.5	Discussion on the Planners Introduced	37
4	Simulation Framework	39
4.1	Simulated Exploration Vehicle Model Dynamics	39
4.2	Generating a Target Field	40
4.3	Simulation Environment	41
5	Results	45
5.1	Zig-Zag Path	45
5.1.1	Simulation Results	46
5.1.2	Simulation Result Parameters	47
5.1.3	Prediction Error Calculation	47
5.2	High Spatial Autocorrelation Results	48
5.2.1	10% Maximum Field Scan	49
5.2.2	20% Maximum Field Scan	51
5.2.3	30% Maximum Field Scan	53
5.3	Half Width Spatial Autocorrelation Results	54
5.3.1	10% Maximum Field Scan	55
5.3.2	20% Maximum Field Scan	57
5.3.3	30% Maximum Field Scan	59
5.4	Quarter Width Spatial Autocorrelation Results	60
5.4.1	10% Maximum Field Scan	61
5.4.2	20% Maximum Field Scan	63
5.4.3	30% Maximum Field Scan	65
5.5	Low Spatial Autocorrelation Results	66
5.5.1	10% Maximum Field Scan	67
5.5.2	20% Maximum Field Scan	69
5.5.3	30% Maximum Field Scan	71

End Matter	73
Bibliography	76
A Additional Results	79
A.1 High Spatial Autocorrelation Results	79
A.1.1 10% Maximum Field Scan	80
A.1.2 20% Maximum Field Scan	82
A.1.3 30% Maximum Field Scan	84
A.2 Half Width Spatial Autocorrelation Results	85
A.2.1 10% Maximum Field Scan	86
A.2.2 20% Maximum Field Scan	88
A.2.3 30% Maximum Field Scan	90
A.3 Quarter Width Spatial Autocorrelation Results	91
A.3.1 10% Maximum Field Scan	92
A.3.2 20% Maximum Field Scan	94
A.3.3 30% Maximum Field Scan	96
A.4 Low Spatial Autocorrelation Results	97
A.4.1 10% Maximum Field Scan	98
A.4.2 20% Maximum Field Scan	100
A.4.3 30% Maximum Field Scan	102

List of Figures

2.1	A Gaussian distributed randomly generated spatially autocorrelated field.	14
2.2	An inverse distance weighting predicted field generated from the samples taken of Figure 2.1a at the locations marked in Figure 2.1b.	15
2.3	An empirical semivariogram.	19
2.4	Examples of three different variogram models.	21
2.5	An experimental variogram generated using Equation 2.7 from the samples taken in Figure 2.1b. δ was chosen such that for n observations, a total number of $\left\lfloor \frac{n}{2} \right\rfloor$ points were plotted. A Gaussian statistical model was fit to the experimental variogram. The variogram was fit using <i>fminsearchcon</i> in MATLAB.	22
2.6	A Kriging Method predicted field generated from the samples taken of Figure 2.1a at the locations marked in Figure 2.1b.	26
3.1	The Next Highest Variance (N HV) algorithm terminated after two iterations of the algorithm. The actual field that is being explored is shown in the upper left figure panel. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle's path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters.	33
3.2	Monte Carlo paths (black) surrounding deterministic N-NHV paths (red). The starting point, ($s = [50 \ 6]^T$), is indicated in green. The set K_N contains $N = 2$ endpoints ($K_N = [5 \ 80]^T, [95 \ 90]^T$). $M_{mc} = 15$ random walks are generated for each endpoint. $\alpha = 5$. The variance of the Wiener process states are $\frac{1}{2}\alpha$	36

3.3	The Monte Carlo Path Planner (MCPP) algorithm (Section 3.4 terminated after two iterations of the algorithm. The actual field that is being explored is shown in the upper left. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle's path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters. Here, N , the number of endpoints selected is $N = 3$, and the number of random paths per cord calculated is 15.	37
4.1	A field is generated using a zero mean normal distribution with variance 1. Varying degrees of spatial autocorrelation are shown for different values of σ_{field}	41
4.2	Next Highest Variance Path Planning. The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.	42
4.3	Monte Carlo path planner run ($N = 5$, $M_{mc} = 20$). The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.	43
4.4	Zig-Zag method. The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.	44
5.1	Zig-Zag exploration method set to scan a fixed percentage of a target field. A Kriging prediction and variance calculation is computed after completing the maneuver. The actual field that is being explored is shown in the upper left. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle's path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters.. .	46
5.2	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	49
5.3	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	50
5.4	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	51
5.5	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	52

5.6	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	53
5.7	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.	54
5.8	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	55
5.9	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	56
5.10	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	57
5.11	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	58
5.12	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	59
5.13	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.	60
5.14	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	61
5.15	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	62
5.16	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	63
5.17	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	64
5.18	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	65
5.19	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.	66
5.20	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	67
5.21	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	68
5.22	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	69
5.23	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	70
5.24	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	71
5.25	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.	72

A.1	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	80
A.2	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	81
A.3	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	82
A.4	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	83
A.5	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	84
A.6	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.	85
A.7	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	86
A.8	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	87
A.9	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	88
A.10	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	89
A.11	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	90
A.12	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.	91
A.13	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	92
A.14	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	93
A.15	A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	94
A.16	Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	95
A.17	A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	96
A.18	Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.	97
A.19	A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	98
A.20	Simulation output for a 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	99

A.21 A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	100
A.22 Simulation output for a 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	101
A.23 A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	102
A.24 Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 3.	103

Abstract

Autonomous Field Exploration via Kriging Prediction Variance Suppression

by

Sargis S Yonan

A set of methods and path planners for the exploration of unknown semi-to-fully-ergodic fields of interest using an autonomous exploration vehicle with turn and speed control are introduced. From observations of a single state of interest, the spatial statistical properties of a target field can be exploited to assist in variance suppressing planning techniques. The Kriging Method, a *Best Linear Unbiased Predictor*, is used to exploit the statistical properties, namely the spatial autocorrelation, of a target field. The Kriging Method predicts the state of unobserved points from a set of observed points for the purposes of quality mapping. A prediction and confidence of prediction of the entirety of a given target field can be generated from the method.

The path planners introduced can be used to reduce the overall prediction uncertainty of a field by optimally steering a single vehicle to collect a good set of samples. A metric for return on investment of executing a trajectory using feedback from Kriging predictions is introduced. The three path planners introduced suppress the overall uncertainty of a Kriging prediction of an unknown target field in order to create a higher quality map when compared to a preplanned scanning regime for the same distance traveled.

TODO

Acknowledgments

TODO

Chapter 1

Introduction

Field exploration is a method in which an unknown field (a *target field*) is learned in an attempt to discover traits or track trends about the field. Field exploration methods can be useful for tracking the health of crop soil, the size of ice glaciers, generating terrain maps, and a wide variety of scientific, agricultural, and industrial purposes. Furthermore, an exploration technique, versus a patrolling or tracking technique where a target is tracked or surveilled, does not require a model of the target field dynamics, as they can be learned on-the-fly. A variety of fields can be explored without the knowledge of the initial values, or model, of the state of interest being observed in the field.

Using an Unmanned Aerial Vehicle (UAV) system, for example, an unknown field of interest can be scanned within a more reasonable time frame compared to conventional scanning techniques involving satellite and manned-airplane missions. Potentially more nuanced data can be gathered from the UAV made observations because of more desirable fields of view and more customizable sensors on-board. Using the techniques introduced,

a high-quality map can be generated of a previously unknown field of interest. Satellite imagery of Earth has been used for measuring various natural phenomena in the past several decades. Estimating polar ice cap melting rates and exploring the location of an oil spills are among the class of problems solved by this technology. Currently, using a service like the US Forest Services' Moderate Resolution Imaging Spectroradiometer (MODIS) Active Fire Mapping Program, images are updated every 1 to 2 days with a fixed sensor. While this program is helpful for detecting large events with long periods of activity, the sampling rate of this service might not give an emergency response team or a scientist the required resolution and precision in gathered data at their desired rate. The resolution and frequency problem along with the cost associated with building, launching, and maintaining an orbiting Earth satellite might even make some areas of research prohibitive. The use of unmanned aerial vehicles (UAVs) have more recently been used in similar fields of study and in environmental protection. The benefits gained from using UAVs is that of more rapidly acquired data with more easily adjusted accuracy. A UAV can give more nuanced and detailed data on features of a field that are not observable from the distance or field of view of an orbiting satellite. This is because the UAV can be equipped with any compatible sensor and can be deployed from virtually anywhere to fly virtually anywhere.

Presently, a common approach to exploring a field is to conduct a zig-zag pattern, or other predetermined maneuver on a target field. This task might take longer than needed to collect the required data, and could potentially ineffectively use the flight or drive time of the exploration vehicle which often has a short and limited runtime. A method that exploits the learned stochastic properties of a field could be used to decrease exploration

time by avoiding the need to scan more points than needed. A majority of the unobserved points in a field can be predicted to a known degree of confidence, given a degree of spatial autocorrelation in the field. Furthermore, scanning every point in a large unknown field is an unrealistic expectation for vehicles with limited maneuvering capabilities. This is especially a problem if the field as a whole is very large and needs to only be predicted to a small degree of confidence. A scheme for minimal and high-quality scanning via variance suppressing path planning would be in the benefit of time for the user(s) of such a system, and the scanning equipment as well.

By exploiting the Kriging variances generated by the predictions from a set of samples taken on a field, variance based path planning methods can be used to steer an exploration vehicle in the areas of maximal uncertainty, while traversing over other areas of low prediction confidence. The methods introduced attempt to help a user of this system explore an unknown field with a known degree of confidence that is configurable through a desired runtime, tuned by the user.

1.1 Previous Works

The goal of this thesis is to introduce path planning techniques which reduce overall uncertainty of Kriging field predictions by steering a single vehicle through a field optimally. Using Kriging predictions as feedback into a path planner, to estimate confidence return for a given candidate trajectory, is introduced in this work.

Previous works have discussed autonomous field exploration in terms of predetermined sampling paths, or slightly dynamically selected paths. The use of Kriging variances in a

path planner in a publication, when this thesis was originally started, did not exist. Near the completion of this thesis, a paper discussing the benefits of using Kriging variance motivated path planning for field exploration, was published [15]. Pulido Fentanes et al. published *Kriging-Based Robotic Exploration for Soil Moisture Mapping Using a Cosmic-Ray Sensor*, where a Kriging variance based exploration technique is used for the purpose of quality mapping of agricultural soil moisture [15]. In the publication, three Kriging variance motivated path planners are used to reduce Kriging prediction error by steering an agricultural robot into areas of high Kriging variance. The first of their path planners, named *Greedy Next-Best-View* (NBV), similar to the *Next Highest Variance* (NHV) strategy demonstrated in this thesis, simply targets the highest variance on the Kriging variance field. In the Greedy NBV algorithm, the path is recalculated every time the robot takes a sample. In NHV, a path is only recalculated when the last endpoint (the highest variance of the last Kriging calculation of the field) is met. The authors of the publication also introduce a *Monte Carlo Next-Best-View* where a set of random endpoints are generated and weighted against one another according to their Kriging variances. The endpoint that yields the point of highest uncertainty is ultimately selected as the next sample location. In this thesis, a Monte Carlo technique is also introduced in Section 3.4, but instead of weighing each proposed random trajectory by its Kriging variance, the predicted values of the points along a random trajectory are plugged back into the sampled field, and a Kriging variance calculation is run on the field again. The path that is ultimately chosen, in this proposed planner, is the path which reduces the expected overall Kriging variance of the field as a whole. Lastly, the authors introduce an adaptive sampling planner which

works by generating an initial path that is then modified after each sample taken. As more possible path are generated randomly, points get removed from the possible set of endpoints when their Kriging variance falls below the mean of the variance field. This method considers the mission time and minimum expectation of the measurement quality by re-planning and using a Traveling Salesperson (TSP) algorithm. In this thesis, minimum expected measurement quality is set before an exploration by tuning the maximum allowed area to scan.

Exploration is a subset of the types of missions UAVs have been used for recently. From Section 2 of Nikhil Nigam’s *The Multiple Unmanned Air Vehicle Persistent Surveillance Problem: A Review* [14], the various types of missions possible are described. There exist problems of tracking and patrolling which involve following a moving target, or of finding the spread rate and source of an item of interest. The exploration mission type is a procedure which runs parallel to the these types of missions. Without a model describing the states of the item of interest being explored, a simple scanning procedure involving random movements or following a predetermined path, like a zig-zag about the field as in [19] are executed, or a zig-zag which incorporates the model dynamics of the vehicle, as in [13]. In *Autonomous Aeromagnetic Surveys Using a Fluxgate Magnetometer* by Douglas G. Macharet et al., A UAV is used in a mineral field exploration technique, where a fluxgate sensor is used to measure the magnetic flux of a vesicle beneath the UAV [5]. A zig-zag pattern is ultimately used to explore the field for minerals of interest. A more dynamic strategy is used in the autonomous home vacuum cleaner *Roomba* by iRobot, where a spiral pattern is used in an attempt to clean up and find the periphery of debris [8]. The radius

of the spiral pattern is a function of the amount of debris tracked by the debris sensor in the immediate area of the vacuum cleaner.

Exploration missions often do not specify the model of the item of interest being tracked. Knowing the model and kinematics of the item being tracked makes it possible to use an optimal estimation tool such as an Extended Kalman Filter as in Rabinovich et al. *A Methodology For Estimation of Ground Phenomena Propagation* [17] and *Multi-UAV Path Coordination Based on Uncertainty Estimation* [16] where the velocity and position states of a ground fire are estimated while tracking the points surrounding the periphery of a wildfire. The planner for this mission calculates a path based off of the Kalman variances of the control points representing the periphery of the ground phenomenon being tracked. Variance based path planning is used in the path planners introduced in this thesis in Chapter 3.

The Kriging Method has been used in a UAV Contour Tracking problem in Zhang et al. *Oil Spills Boundary Tracking Using Universal Kriging And Model Predictive Control By UAV* [24]. The work relies on the knowledge of a model of the oil spill, and therefore is not a generic case of an exploration problem of a model-less field.

C. C. Castello et al. present the use of the Kriging method for environmental sensor placement in *Optimal Sensor Placement Strategy for Environmental Monitoring using Wireless Sensor Networks* [1]. The overall variances of a Kriging predicted field, predicted from a set measurements from fixed sensor locations, can be directly compared to the variances of predicting the same field with a different set of sensor locations. The method can therefore be used to help assist in optimal sensor placement by conducting a Monte Carlo

simulation of random sensor placements, and ultimately choosing the random configuration that minimized the Kriging prediction variances for the field. A path planner, which can be stated as a sensor placement problem, by selecting a random path, or set of sampling locations, that minimizes the expected Kriging variance of a target field is introduced in this thesis in Section 3.4. The use of a Monte Carlo approach, where noise is used to assist in suppressing prediction uncertainty has been used for uncertainty suppression in obstacle avoidance motion planning in *Monte Carlo Motion Planning for Robot Trajectory Optimization Under Uncertainty* [10], but the technique is not used for exploration purposes, as introduced in this thesis.

1.2 Problem Definitions

In an effort to be consistent in naming conventions and parameter definitions throughout this work, the problem space will be defined. The conventions described in Section 1.2 will be used throughout the rest of the work.

1.2.1 The Field

The initially unknown field, referred to as the *target field*, is a rectangular field of height h , and width w , i.e. $Z \in \mathbb{R}^{h \times w}$. The field is made up of square pixel cells, referred to as *vesicles*. Each vesicle can be “visited”, or sampled, in order to yield a single state of interest in the set of real numbers. Throughout this thesis, a square target field (i.e. $h = w$) will be used.

1.2.2 The Sensor

The observations of interest made on the field will be using ideal sensors with no measurement noise. The sensors will measure a subset of the area of the entire target field. This area will be referred to as the *sensor footprint*, and will be equal to the size of a single vesicle of the target field.

For the methods developed, the locations of the sensor measurements must be known. The locations will be represented as Cartesian coordinates on the field. For an arbitrary observation of the field, the location of the measurement will be at ccorrespondingoordinates $s \in \mathbb{R}^2$, and the sensor measurement would be $Z(s)$. The value of Z at s is quantized to the vessicle in which the point s falls within.

Real World Sensing Examples

A Global Positioning System (GPS) sensor would likely be used to estimate localized position of a sensor measurement on Earth. In the case of predicting the boundaries of a glacier, for example, an infrared sensor would likely be used to measure the state of interest, thermal output of the field in this case. In the case of terrain mapping, for example, a LiDAR sensor could be used to sample terrain altitudes of the terrain below, at marked locations using GPS, on a UAV.

Chapter 2

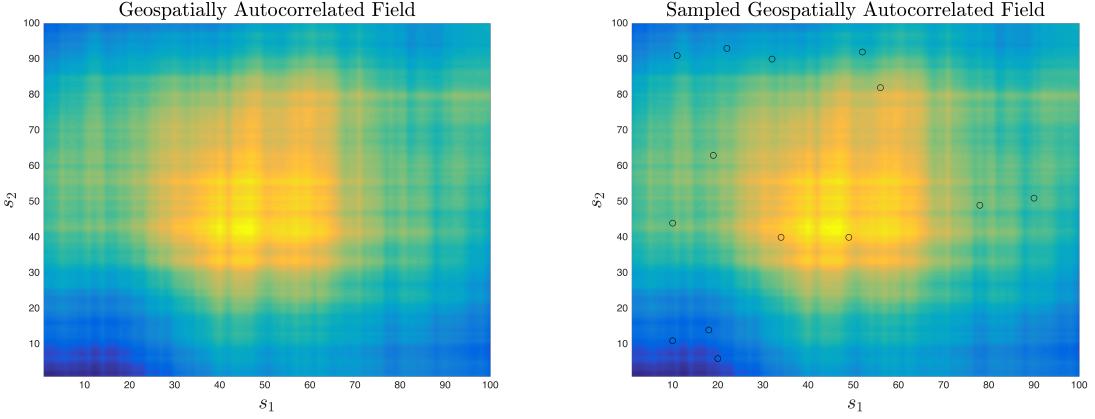
Spatial Analysis

Many of the methods introduced in this thesis will rely on works developed in the fields of Spatial Statistics, Geostatistics, and Geography. Geo-statisticians have developed much of the work surrounding field predictions in the geospatial domain. In an effort to make the methods introduced be as presentable and contained as possible, a background on the required tools from the field of geostatistics will be reviewed. The Kriging Method, a best linear unbiased predictor (BLUP) produces a prediction based on statistical data gathered from samples taken on a field. The method performs a weighted sum generated from a covariance matrix created from points sampled on a field. A variance for each prediction computed can also generated as a byproduct of the Kriging prediction, and will be used to calculate information gain in a path finder introduced in this paper. It is assumed that the expected value of each point is from a normal distribution, where the variance and expected value of the distribution is a function of the neighboring samples and spatial autocorrelation factor of the field.

Tobler’s First Law of Geography [22] states, “Everything is related to everything else, but near things are more related than distant things.” Regarding geospatial data, there is a positive correlation between observations with a small difference in distance [12]. This implies the existence of geospatial autocorrelation in many target fields of interest, where there exists a positive correlation between elements in a field. Geospatial autocorrelation is the hypothesis that allows naive prediction techniques, like Inverse Distance Weighting (IDW) (Section 2.1), to work. The Kriging Method first finds the underlying spatial autocorrelation of a target field from a set of samples, and then predicts the state of a given vesicle by emphasizing values of statistically similar samples in a weighted sum. The methods introduced in this chapter are intended to serve as an introduction and background into the Kriging Method.

2.0.1 Autocorrelation in a Field

Positively correlated spatial autocorrelation in a field implies the existence of a cluster of similar points near one another i.e. relatively small covariances between two spatially similar points. The opposite is true when the overall spatial autocorrelation of a field is negative. Using Tobler’s First Law of Geography, the assumption that fields measured will contain positive autocorrelation will be used. The degree of spatial autocorrelation in a field can be measured, and will be discussed in Section 2.2 on Variography.



(a) A randomly generated spatially autocorrelated field.
(b) Samples at marked locations were taken of the target field in 2.1a.

Figure 2.1: A Gaussian distributed randomly generated spatially autocorrelated field.

2.1 Inverse Distance Weighting

An inverse distance weighting is a naive interpolation tool where a point is predicted based its distances from a set of observed points. A simple IDW, using Shepard's Method [20], gives a prediction, $\hat{Z}(\mathbf{s}_j)$, of an unobserved point, \mathbf{s}_j , as a function of the $N \in \mathbb{N}$ observed points, $\{Z(\mathbf{s}_1), Z(\mathbf{s}_2), \dots, Z(\mathbf{s}_n)\}$.

$$\hat{Z}(\mathbf{s}_j) = \begin{cases} \frac{\sum\limits_{i=1}^N [w(\mathbf{s}_j, \mathbf{s}_i)] Z(\mathbf{s}_i)}{\sum\limits_{i=1}^N w(\mathbf{s}_j, \mathbf{s}_i)} & \text{if } \forall i \mid d(\mathbf{s}_j, \mathbf{s}_i) \neq 0 \\ Z(\mathbf{s}_j) & \text{if } \exists i \mid d(\mathbf{s}_j, \mathbf{s}_i) = 0 \end{cases} \quad (2.1)$$

$$d(\mathbf{s}_j, \mathbf{s}_i) = \|\mathbf{s}_j - \mathbf{s}_i\|_2 \quad (2.2)$$

$$w(\mathbf{s}_j, \mathbf{s}_i) = \frac{1}{d(\mathbf{s}_j, \mathbf{s}_i)^p} = \|\mathbf{s}_j - \mathbf{s}_i\|_2^{-p} \quad (2.3)$$

where $p \in \mathbb{R}^+$ is the IDW "power parameter". The power parameter, p , controls the emphasis on near and far observations on a prediction. As p increases, the predicted values more closely resemble the closest made observation to the prediction location. Inversely,

as p gets smaller within $(0, 1]$, more emphasis is drawn from observations made further away.

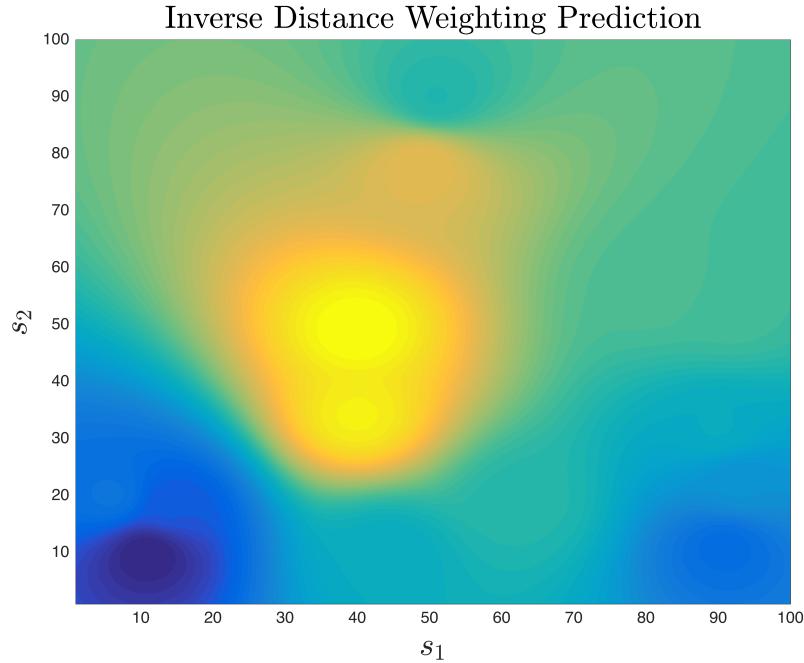


Figure 2.2: An inverse distance weighting predicted field generated from the samples taken of Figure 2.1a at the locations marked in Figure 2.1b.

This method can yield a prediction for all possible s_j points in a field where a set of observations at known locations are made, as done in Figure 2.2. Unfortunately, the method is limited in that it does not take advantage of the underlying stochastic model, and spatial pattern, of the target field being observed to make a more methodical weighted sum prediction. A field that exhibits properties of spatial autocorrelation would be more statistically exploitable because the distribution of the states of interest on the field can be learned.

2.2 Variography

Variography is a set of procedures for examining and interpreting spatial dependence and spatial autocorrelation in a field of observed data. In order to make a more intelligent weighted sum for a prediction, extracting the underlying spatial autocorrelation function, the *variogram*, of a field will be introduced. The variogram function will be factored into a classical prediction via weighting, yielding a Kriging Weighting.

2.2.1 The Variogram

A variogram quantifies dependence for two disjoint observations separated by some distance, or *lag*, away. The function, in essence, yields a value directly proportional to the covariance between two given points in a stochastic field.

A Variogram is intended to be a continuous function which yields a covariance between two points $Z(\mathbf{s}_i)$, $Z(\mathbf{s}_j)$, which have not necessarily been observed, but known to be a Euclidean distance, or lag, $h_{i,j} \in \mathbb{R}$ apart [4], where

$$h_{i,j} = \|\mathbf{s}_i - \mathbf{s}_j\|_2 \quad (2.4)$$

Using the assumption on what a point's value on a field is constructed of is made in Equation 2.4.1 of Matheron, 1963 [11]:

$$Z(\mathbf{s}_i) = \mu(\mathbf{s}_i) + \theta(\mathbf{s}_i) \quad (2.5)$$

where $\theta(\cdot)$ is a zero-mean intrinsically stationary stochastic Wiener process. An assumption that the mean $\mu(\cdot) = \bar{Z}$ is only constant in a reasonably small neighborhood of Z . This becomes relevant when the sample sizes of the field increase where more reliable means will

be derived from local neighborhoods. The size of the local neighborhoods to consider a constant field mean within will later be defined to be a function of the maximum autocorrelation lag, but no cut-off is required by definition.

2.2.2 The Semivariogram

The Semivariogram is defined to be the average squared difference between two points separated by some distance apart. Matheron, 1963 formally defines a variogram in [11] in three-dimensional space. Using the notation used in this paper for a two-dimensional field, the Semivariogram will be defined as:

$$\gamma(h) = \frac{1}{2A} \iint_A [Z(\mathbf{s} + h) - Z(\mathbf{s})]^2 dA \quad (2.6)$$

where A is a closed area in a field to consider, $Z(\mathbf{s})$ is the value of a point at location \mathbf{s} on the field, and $Z(\mathbf{s} + h)$ is the value of some point a distance h , defined in Equation 2.4, apart from a point \mathbf{s} on the field.

It is infeasible to estimate an observation value at each possible point in the field to compute a continuous Semivariogram. Furthermore, the fields observed using these methods are typically gridded, and therefore not continuous by their analytical nature. A discrete model must first be constructed, and will then be fit into a continuous variogram model. This is done by first constructing a discrete variogram model, or *Empirical Semivariogram*, and then fitting a continuous model to it. Fitting a discrete Semivariogram should in turn yield a function close to $\gamma(h)$ defined in Equation 2.6, and should be identical given that every point in the area A is sampled with infinite precision.

2.2.3 The Empirical Semivariogram

An Empirical Semivariogram, or Experimental Semivariogram, is a discrete function representing the covariance of the observation value difference between two sampled locations that are some distance h apart. Goovaerts defines the empirical variogram in *Geostatistics for Natural Resources Evaluation. Applied Geostatistics Series* [21] as:

$$\hat{\gamma}(h) = \frac{1}{2N(h)} \sum_{i=1}^{N(h)} (Z(\mathbf{s}_i) - Z(\mathbf{s}_i + h))^2 \quad (2.7)$$

where $N(h)$ is the cardinality of the set of all pairs of observed points that are a Euclidean distance, or lag, h , apart.

The experimental variogram conveys the spatial autocorrelation of a sampled field. As the lag between two given points increases, the covariance also increases when the field is spatially autocorrelated. The covariance levels out to a steady value (the *sill*) at some distance in the domain (the *range*). This position in the function marks where the loss of reliable spatial autocorrelation between two points that are a distance h apart lays.

2.2.4 Converting a Semivariogram to a Variogram

The intent of fitting a statistical model to an experimental variogram is to approximate the continuous covariance for any two points, that have not necessarily been observed, on Z that are at some known lag apart.

2.2.5 Variogram Models

The Empirical Semivariogram will be fit to a statistical model, or *kernel*, known as a Variogram Model. There exist some well known models, further discussed in this section.

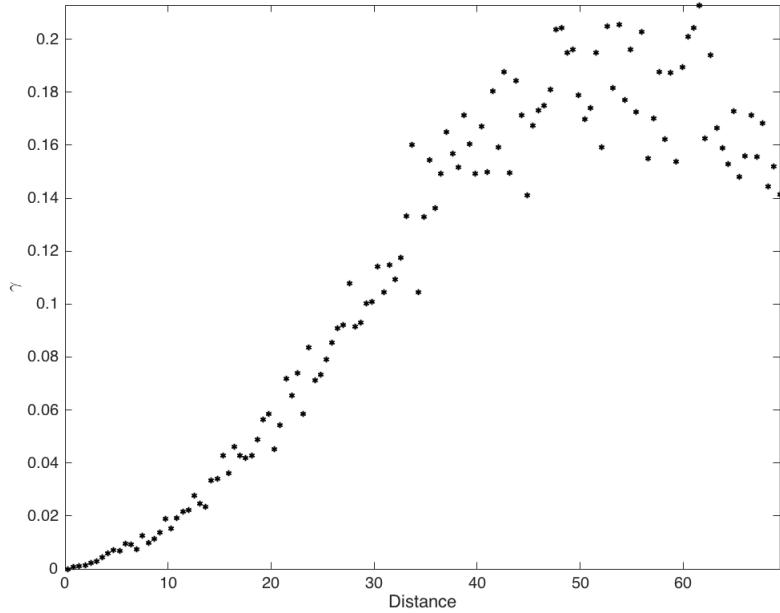


Figure 2.3: An empirical semivariogram.

Each model is a scalar function of lag, h , sill, s , and range, a . The term *sill* refers to the point on the co-domain where two points at the lag specified are no longer autocorrelated. The sill is therefore the largest value of covariance for two disjoint points on a field that are still considered to be autocorrelated. The corresponding point on the domain for the sill is referred to as the *range* on the variogram. Two points that have a lag larger than the range are not considered to be autocorrelated.

The *nugget* of the variogram is defined to be the variance at zero separation, or $\gamma(0)$ [11]. This value is exactly zero for ideal measurements, but is generally not for real-life measurements. The value found for the nugget is typically summed with the value yielded by γ , to get the final variogram value for a given lag [21].

The Gaussian Model

$$\gamma_g(h, s, a) = s \left[1 - \exp \left(-\frac{h^2}{a^2} \right) \right] \quad (2.8)$$

The Gaussian model will asymptotically reach its sill. The sill would be at the limit as h approaches infinity. The *practical range* is therefore used to refer the point on the domain where the variogram reaches 95% of its sill [21].

The Exponential Model

$$\gamma_e(h, s, a) = s \left[1 - \exp \left(-\frac{h}{a} \right) \right] \quad (2.9)$$

The same rules as the Gaussian model apply to the Exponential model [21].

The Spherical Model

$$\gamma_s(h, s, a) = \frac{s}{2} \left[\frac{3h}{a} - \left(\frac{h}{a} \right)^3 \right] \quad (2.10)$$

The spherical model will reach an exactly zero slope at the sill and range [21].

2.2.6 Fitting A Semi-Variogram

The kernel function of the range, a , the sill, s , and lag, h is chosen based on the statistical properties of the field being examined. Although there exist no closed form solution for finding an appropriate variogram model for a given field, one can compare a variety of different models against one another. Conducting cross-validation tests and comparing root-mean squared prediction errors for different models are common approaches for finding appropriate variogram models.

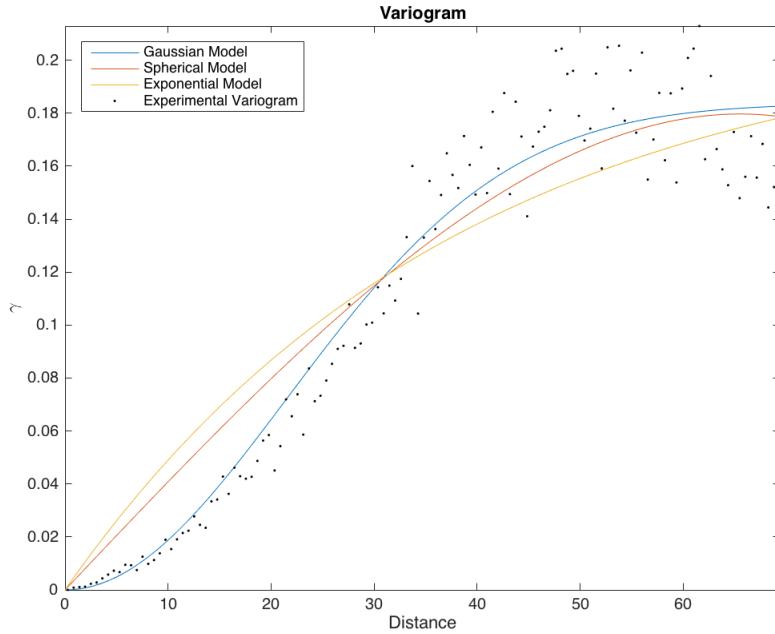


Figure 2.4: Examples of three different variogram models.

Fitting a Variogram Using MATLAB

Using a version of the *fminsearch* function in *MATLAB*, a variogram can be fit to the desired objective function from a set of samples and initial guesses for the range and sill using a simplex search method. As the function is used over several iterations of sampling, the fit range and sill values found in the previous iteration can be used as the seed to the next iteration of the fit in an attempt to minimize computation time. The *MATLAB* function, *fminsearch*, is defined to “find the minimum of an unconstrained multi-variable function using a derivative-free method” [9], expressed in Equation 2.11.

$$\gamma(h) = \min [\gamma_{kernel}(h, s, a) - \hat{\gamma}(h)]^2 \quad (2.11)$$

The function is then modified by specifying bounds of minimization in an attempt to decrease iterations of the function fit, which can be computationally expensive as more

samples are taken. This modified version of *fminsearch*, named *fminsearchcon*, can be downloaded from the MathWorks File Exchange.

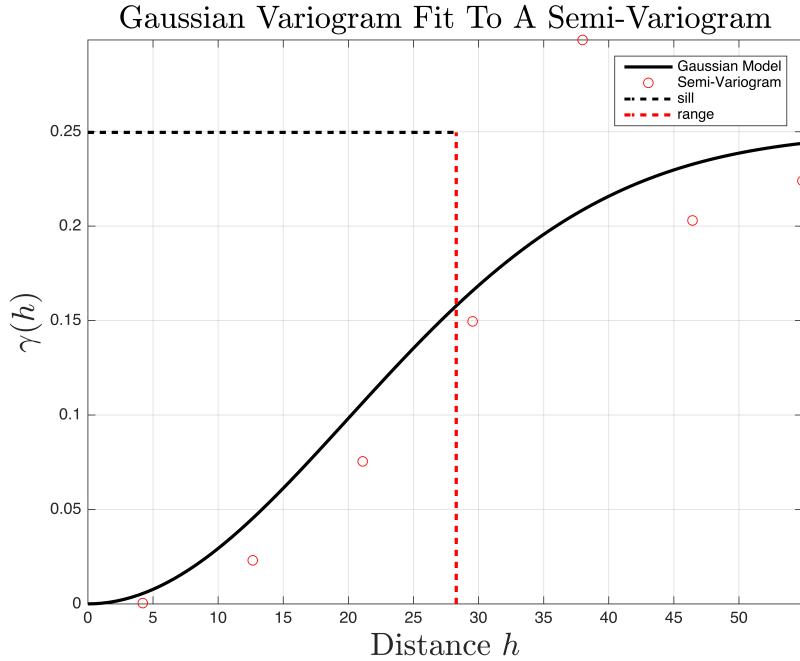


Figure 2.5: An experimental variogram generated using Equation 2.7 from the samples taken in Figure 2.1b. δ was chosen such that for n observations, a total number of $\left\lfloor \frac{n}{2} \right\rfloor$ points were plotted. A Gaussian statistical model was fit to the experimental variogram. The variogram was fit using *fminsearchcon* in MATLAB.

2.3 The Kriging Method

The Kriging Method conducts a weighted sum using the continuous variogram model that was fit to the physical observations made. The method can yield a prediction for each vesicle in a target space similar to the Inverse Distance Weighting method described in Section 2.1, but with more statistical robustness.

2.3.1 Forms of the Kriging Method

There exist three major forms of the Kriging Method. All of which differ primarily in the handling of the mean gathered from observations of a target field. The *Simple Kriging Method* makes the assumption that the mean is known and constant throughout the entirety of an observed field. This is of course not the case for fields that are very large as it does not follow Tobler's First Law. The *Ordinary Kriging Method* can deduce the local mean of a neighborhood from a smaller subset of observations in a larger target field. This is done by classifying the larger field into smaller neighborhoods where the mean is only constant within those neighborhoods. Ordinary Kriging has the advantage that the mean is not required to be known before running a prediction. The *Universal Kriging Method* can perform similar local mean calculations as the Ordinary Kriging Method, but does so by fitting a polynomial representing a mean trend model and not from a constant mean value representing that neighborhood [23] as seen in Section 2.2.6 on fitting a variogram.

2.3.2 Covariance Matrix From A Variogram

From the fit variogram which represents the spatial statistics of a field from a set of samples, a variance-covariance matrix for N observations, $P \in \mathbb{R}^{N+1 \times N+1}$, will be constructed. The value of the element $P_{i,j}$, will represent the covariance of the lag between the i^{th} and j^{th} observations on the field [21], [11]. If $i = j$, the value of the element, $P_{i,j}$ is the variance of the i^{th} observation.

$$P_{i,j} = \text{cov}\{Z(\mathbf{s}_i), Z(\mathbf{s}_j)\} = \gamma(\|\mathbf{s}_i - \mathbf{s}_j\|_2) \quad (2.12)$$

$$P = \begin{bmatrix} \gamma(0) & \gamma(\|\mathbf{s}_1 - \mathbf{s}_2\|_2) & \dots & \gamma(\|\mathbf{s}_1 - \mathbf{s}_N\|_2) \\ \gamma(\|\mathbf{s}_2 - \mathbf{s}_1\|_2) & \gamma(0) & \dots & \gamma(\|\mathbf{s}_2 - \mathbf{s}_j\|_N) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(\|\mathbf{s}_N - \mathbf{s}_1\|_2) & \gamma(\|\mathbf{s}_N - \mathbf{s}_2\|_2) & \dots & \gamma(0) \end{bmatrix} \quad (2.13)$$

2.3.3 The Proximity Vector

For any given point on a field, we can construct a *proximity vector*, $\mathbf{d}_0 \in \mathbb{R}^{N+1}$, which contains the covariance of a given point, \mathbf{s}_0 on the field with the N observations made. The k^{th} element of \mathbf{d}_N , would therefore contain the covariance for the lag between point \mathbf{s}_0 and the k^{th} observation made, \mathbf{s}_k [11].

$$\mathbf{d}_0(k) = \gamma(\|\mathbf{s}_0 - \mathbf{s}_k\|_2)$$

$$\mathbf{d}_0 = \begin{bmatrix} \gamma(\|\mathbf{s}_0 - \mathbf{s}_1\|_2) \\ \gamma(\|\mathbf{s}_0 - \mathbf{s}_2\|_2) \\ \vdots \\ \gamma(\|\mathbf{s}_0 - \mathbf{s}_N\|_2) \end{bmatrix} \quad (2.14)$$

Furthermore, The Kriging Method can be bounded. If a to-be-predicted point and a given sample is beyond the range value fit to the variogram model, the corresponding element in the proximity vector is set to the sill. This ensures that points outside of the range of autocorrelation are not weighted anymore than they should be. This method is suggested when the variogram model used is a bounded function, e.g. the Spherical Model (Equation 2.10).

2.3.4 The Kriging Weights

Similarly to the Inverse Distance Weighting method, a set of weights will be computed for each vesicle in the target field. These weights will be referred to as the *Kriging Weights*. For a given prediction location, s_0 , the Kriging Weight vector, λ_0 , will be defined as the product of the inverse of the covariance matrix of the field and the proximity vector of the point to predict [6].

$$\begin{bmatrix} \lambda_0 \\ \mu_0 \end{bmatrix} = \begin{bmatrix} P^{-1} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} d_0 \\ 1 \end{bmatrix} \quad (2.15)$$

where μ_0 is the Lagrangian multiplier that is used to assist in maintaining the unbiasedness condition of the Kriging method [6].

2.3.5 The Kriging Prediction Equation

The Kriging equation will be used to predict the value, $\hat{Z}(s_0)$ of an unobserved location, s_0 . The prediction is a function of the Kriging Weights and a vector of N observations [6].

$$\hat{Z}(s_0) = [Z(s_1) \ Z(s_2) \ \dots \ Z(s_N)] \lambda_0 \quad (2.16)$$

2.3.6 Variance of A Kriging Prediction

The variance of a point predicted on a target field can be calculated using byproduct terms generated along the way of calculating a Kriging prediction [6]. For a predicted point $\hat{Z}(s_0)$, using the proximity vector, d_0 , defined in Section 2.3.3, and the Kriging Weights, λ_0 defined in Section 2.3.4 for the predicted point, the variance of the prediction for that point is defined as:

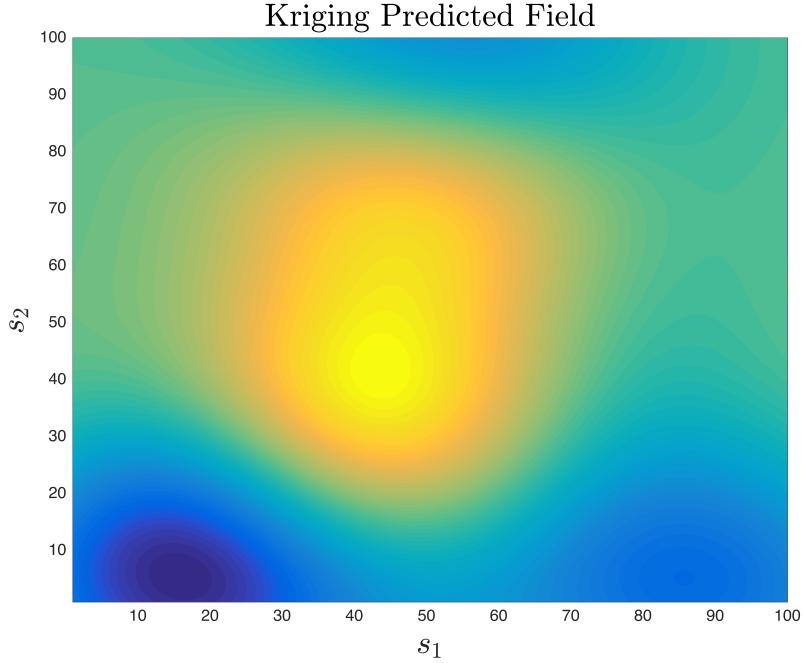


Figure 2.6: A Kriging Method predicted field generated from the samples taken of Figure 2.1a at the locations marked in Figure 2.1b.

$$\text{var}\{\hat{Z}(s_0)\} = \begin{bmatrix} \mathbf{d}_0 \\ 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_0^T & \mu_0 \end{bmatrix} \quad (2.17)$$

2.3.7 Procedure For Field Prediction Using The Kriging Method

In order to predict the entirety of a target field from a finite set of N observations and their respective locations, O , the Kriging Prediction is run at every possible unobserved vesicle in the target field. For a single iteration of collecting observations and making predictions, a covariance matrix must first be constructed, and then a the proximity vector and Kriging Weights are computed for all unobserved vesicles. The Kriging prediction formula is then used to compute the predicted value of each vesicle to predict.

When Algorithm 1 is run on the target field from Figure 2.1a, for the samples taken in

Algorithm 1: Kriging Prediction of Target Field

```
1: procedure KRIGINGPREDICTFIELD( $Z, O$ )
2:   Generate Semi-Variogram:
3:      $\forall \mathbf{s}_i, Z(\mathbf{s}_i) \in O$ :
4:        $\hat{\gamma}(h) \leftarrow \mathbf{s}_i, Z(\mathbf{s}_i)$ 
5:
6:   Generate Variogram:
7:      $\gamma(h)$  fits to  $\hat{\gamma}(h)$ 
8:
9:   Construct Covariance Matrix:
10:     $\forall (\mathbf{s}_i, \mathbf{s}_j) \in O :$ 
11:       $h_{i,j} = \|\mathbf{s}_i - \mathbf{s}_j\|_2$ 
12:       $P_{i,j} = \gamma(h_{i,j})$ 
13:
14:     $\forall i \in [1, N] :$ 
15:       $P_{i,N+1} = 1$ 
16:       $P_{N+1,i} = 1$ 
17:       $P_{N+1,N+1} = 0$ 
18:
19:   Run Kriging Predictions For Target Field:
20:    $\forall \mathbf{p}_i \in field:$ 
21:      $\mathbf{d}_i = \begin{bmatrix} \gamma(\|\mathbf{s}_1 - \mathbf{p}_i\|_2) \dots \gamma(\|\mathbf{s}_N - \mathbf{p}_i\|_2) & 1 \end{bmatrix}^T$ 
22:      $\begin{bmatrix} \boldsymbol{\lambda}_{\mathbf{p}_i} \\ \mu_{\mathbf{p}_i} \end{bmatrix} = \begin{bmatrix} P^{-1} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{d}_{\mathbf{p}_i} \\ 1 \end{bmatrix}$ 
23:      $\hat{Z}(\mathbf{p}_i) = \begin{bmatrix} Z(\mathbf{s}_1) \dots Z(\mathbf{s}_N) \end{bmatrix} \boldsymbol{\lambda}_{\mathbf{p}_i}$ 
24:      $\text{var}\{\hat{Z}(\mathbf{p}_i)\} = \begin{bmatrix} \mathbf{d}_i \\ 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_{\mathbf{p}_i}^T & \mu_{\mathbf{p}_i} \end{bmatrix}$ 
```

Figure 2.1b, a prediction of the entire field can be generated, as seen in Figure 2.6.

Chapter 3

Path Planning

The goal of each of the planners introduced is to assist in the discovery of a field's features with an adjustable trade off between speed and confidence of prediction. The user of such a system could choose to scan more area if fuel is not of high concern. Likewise, if the field is very large, or several fields need to be scanned in a limited amount of time, a quicker scan with a lower degree of prediction certainty can be performed.

3.1 Vehicle & Information Gain Model

In an effort to formalize the dynamics of the exploration vehicle, and the information gain on the field, the models for vehicle dynamics and field variances will be introduced.

3.1.1 Exploration Vehicle Model Dynamics

The state vector of the vehicle will be defined as follows:

$$\mathbf{X} = \begin{bmatrix} x \\ y \\ \theta \\ V \end{bmatrix} \quad (3.1)$$

where x and y are the vehicle's position on a field, θ is the vehicle's heading angle, ω is the vehicle's angular velocity, $\dot{\theta}$, and V is the magnitude of the linear velocity of the vehicle. Both ω and V are control inputs to the vehicle.

The exploration vehicle dynamics will be modeled after a simple forward discrete kinematics model with a constant time step per iteration, ΔT . An iteration of the propagation model will be the sum of the previous iteration, \mathbf{X}_k , the nonlinear vehicle dynamics, $\mathbf{f}(\mathbf{X}_k)$, and the control input, \mathbf{u}_k .

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{f}(\mathbf{X}_k) + \mathbf{u}_k \quad (3.2)$$

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \begin{bmatrix} V_k \Delta T \cos \theta_k \\ V_k \Delta T \sin \theta_k \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \theta_k \\ V_k \end{bmatrix} \quad (3.3)$$

The speed, V , is assumed to be regulated at a constant value for all values of k .

3.1.2 Field Uncertainty Model

In Section 2.3.6, a method for calculating the variance of a prediction was defined as a function of the proximity vector and Kriging weights generated for the prediction point. For points that have been directly measured, the variance is ideally zero (for fields with no drift or dynamics). The uncertainty of the prediction of a point in the target field is directly proportional to the variance of its prediction. The goal of a path planner intending

to suppress uncertainty of predictions in a target field would be to reduced the overall variance of the target field being explored.

A criterion for overall field uncertainty can be defined as the average variance, calculated from a prediction of a target field from a set S points sampled for all $h \times w$ predictable points on a target field.

$$\Sigma(\hat{Z}_S) = \frac{1}{hw} \sum_{i=1}^{hw} \text{var}\{\hat{Z}(\mathbf{p}_i)\} \quad (3.4)$$

3.1.3 Uncertainty Loss Function

In Section 3.1.2, a criterion for overall field uncertainty was introduced. Given a set of sampled points, S on a field, the overall field uncertainty is the mean variance of all points on the field, $\Sigma(\hat{Z}_S)$. For an additional set of samples, T , taken on the field, a new field uncertainty, $\Sigma(\hat{Z}_{S \cup T})$, is the field uncertainty criterion of the fields prediction from the union of the sample sets S and T . The difference in overall field uncertainty, $L(T)$, will be defined as the uncertainty lost by taking the additional samples in the set T on the field.

$$L(T) = \Sigma(\hat{Z}_S) - \Sigma(\hat{Z}_{S \cup T}) \quad (3.5)$$

3.2 Next Highest Variance Path Planner

Sampling the location of the next highest variance (NHV) is the simplest and most naive approach to path planning using the Kriging method. The highest point of uncertainty on the field is the point, \mathbf{p} , is defined as:

$$\arg \max_{\mathbf{p}} \text{var}\{\hat{Z}(\mathbf{p})\} \quad (3.6)$$

By simply setting the next endpoint of the path to \mathbf{p} , the point of highest uncertainty will be sampled at the end of the path. Once the point is met at the end of the path, a new set of samples gathered from the path to the endpoint will be used to recalculate the statistical patterns of the field to higher degree of quality. A Kriging prediction, variances of those predictions are then run on the field. The path planner continues by setting the next endpoint to the next point of highest uncertainty after recalculating the variances of the field. The planner terminates exploration once a preset maximum scan area limit has been met by the exploration vehicle. An initial forward diagonal sweep of samples are taken across the field to generate an initial variogram and Kriging prediction variances before the path planner chooses its first endpoint.

3.2.1 Inefficiency in NHV

The NHV algorithm does not account for repeating paths, or avoiding the re-sampling of points on the field. The only knowledge used is the variance of the endpoint of a path. Although the ground covered by the algorithm may be sufficient for uncertainty suppression, a path planner that considers the cost of trajectories would likely yield better results.

3.3 N Next Highest Variances Path Planner

A modification to the NHV algorithm can be made to consider more trajectories, and therefore compare the loss of taking a path from a set of possible paths. Let K_N be the set of the $N \in \mathbb{N}$ points of highest uncertainty on the field. Let T_i be a candidate trajectory connecting the current position of the exploration vehicle to the i^{th} point in the set K_N .

The endpoint that is ultimately chosen by the N Next Highest Variance (N-NHV) Path Planner is the one that maximizes the loss function, $L(T_i)$. The points along a trajectory, T_i , have likely not been sampled, as they represent points of high uncertainty. The loss in uncertainty for taking the path, T_i , is therefore not known. An estimate of the loss in overall field uncertainty, $\hat{L}(T_i)$, after taking the path, T_i , is calculated by using the previous Kriging predictions of the points along the path. The predictions of those points are used as actual samples taken on the field in a new Kriging prediction variance calculation of the

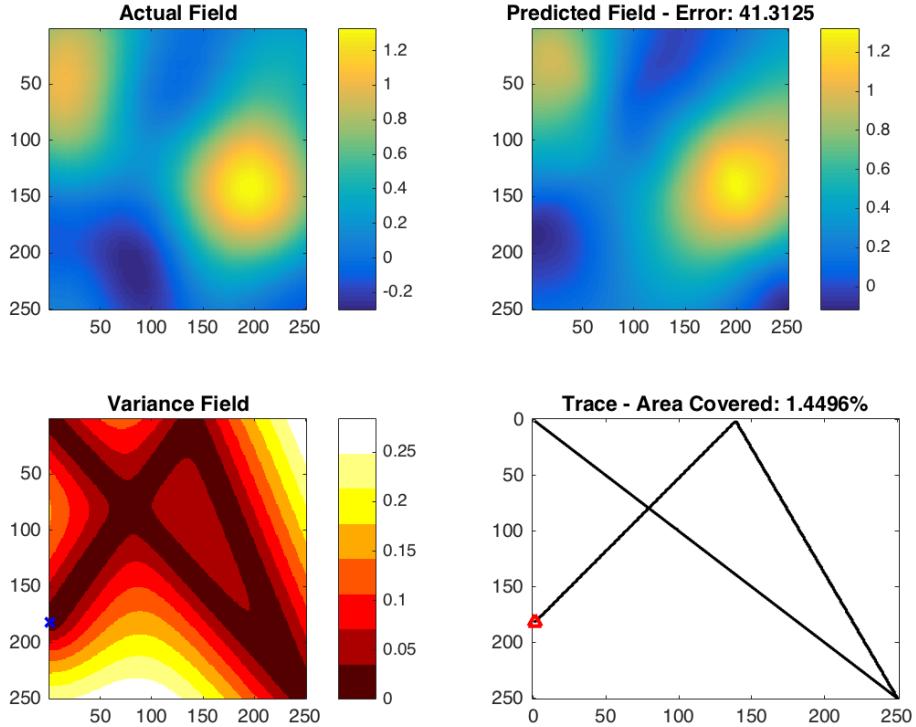


Figure 3.1: The Next Highest Variance (N HV) algorithm terminated after two iterations of the algorithm. The actual field that is being explored is shown in the upper left figure panel. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle's path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters.

field. An initial forward diagonal sweep of samples are taken across the field to generate an initial variogram, initial Kriging predictions, and initial Kriging prediction variances before the path planner chooses its first endpoint.

3.4 Monte Carlo Path Planner

A Monte Carlo path is one where at each step in each of the candidate trajectories generated from the set of candidate endpoints, K_N , a random movement is made. The path has an expected value which is equal to the original, non-stochastic, N-NHV version of the path generated for the equivalent point in K_N . By introducing noise into each of the trajectories found in the N-NHV path planner, a more optimal path can be found. Similarly to the N-NHV method, the Monte Carlo path that maximizes the loss of the overall field uncertainty is chosen as the next vehicle path. Let K_N be the set of the N points of highest prediction variances on the field. Let T_i be a candidate trajectory from the current position of the exploration vehicle to the i^{th} endpoint in the candidate endpoint set, K_N . Each point in the candidate trajectory is a waypoint the exploration vehicle will visit on its way to the last point in the sequence. The trajectory is a set of states representing the field horizontal position, x , the field vertical position, y , and the vehicle heading angle, θ . The k^{th} vector in the state candidate trajectory set, T_i , denoted as $T_i(k)$, will be the state the exploration vehicle will take on at that position on the field, i.e.

$$T_i(k) = \begin{bmatrix} x_i(k) \\ y_i(k) \\ \theta_i(k) \end{bmatrix} \quad (3.7)$$

Let $\alpha \in \mathbb{R}$ be the step size of the vehicle from one point to the next within the trajectory,

T_i . Let $\mathbf{w}_i \in \mathbb{R}^2$ be a vector of two zero-mean Wiener processes with a tunable process standard deviation which is less than the step size, α . Furthermore, the step size of the vehicle, α , can not be a value less than the distance the exploration vehicle can travel in one time-step, ΔT , given a constant vehicle velocity of V .

$$\text{var}\{\mathbf{w}_i\} = \begin{bmatrix} \text{var}\{w_{i_x}\} \\ \text{var}\{w_{i_y}\} \end{bmatrix} \quad (3.8)$$

$$\text{var}\{w_{i_x}\} < \alpha^2 \quad (3.9)$$

$$\text{var}\{w_{i_y}\} < \alpha^2 \quad (3.10)$$

$$\alpha \geq V\Delta T \quad (3.11)$$

The corresponding Monte Carlo path, or sequence of way-points the exploration vehicle will make on its way to the candidate endpoint, $\mathbf{p}_i = [p_x \ p_y]^T$.

$$T_i(k) = \begin{cases} \begin{bmatrix} x_0 \\ y_0 \\ \text{atan2}(p_y - y_0, p_x - x_0) \end{bmatrix} & : k = 1 \\ \begin{bmatrix} \alpha \cos \theta_i(k) \\ \alpha \sin \theta_i(k) \\ \text{atan2}(p_y - y_i(k), p_x - x_i(k)) \end{bmatrix} + \begin{bmatrix} \mathbf{w}_i(k) \\ 0 \end{bmatrix} & : 1 < k < \left\lceil \frac{\|\mathbf{p}-\mathbf{s}\|_2}{\alpha} \right\rceil \\ \begin{bmatrix} p_x \\ p_y \\ 0 \end{bmatrix} & : k = \left\lceil \frac{\|\mathbf{p}-\mathbf{s}\|_2}{\alpha} \right\rceil \end{cases} \quad (3.12)$$

where the initial point in the candidate trajectory is set to the current position, $\mathbf{s} = [x_0 \ y_0]^T$ from the state vector of the exploration vehicle. Due to the uncertainty in length of the random trajectory generated, the number of waypoints in the candidate trajectory T_i is fixed, such that $k \in [1, \left\lceil \frac{\|\mathbf{p}-\mathbf{s}\|_2}{\alpha} \right\rceil]$ (the number of points in the trajectory for a step size, α ,

given zero variance noise added to the process). The last point in the candidate trajectory set is set to the corresponding endpoint from the set K_N .

As introduced in the N-NHV path planner, a candidate trajectory is generated for each of the candidate endpoints in the set, K_N . The candidate trajectory that maximizes the loss function, $L(T)$, is the path that is ultimately selected as the next vehicle trajectory. In an effort to find a more optimal trajectory, more than one random walk can be generated for each endpoint in the set K_N . The variable $M_{mc} \in \mathbb{N}$ will denote the number of random walks taken per endpoint in the set K_N . An initial forward diagonal sweep of samples are taken across the field to generate an initial variogram, initial Kriging predictions, and initial Kriging prediction variances before the path planner chooses its first endpoint.

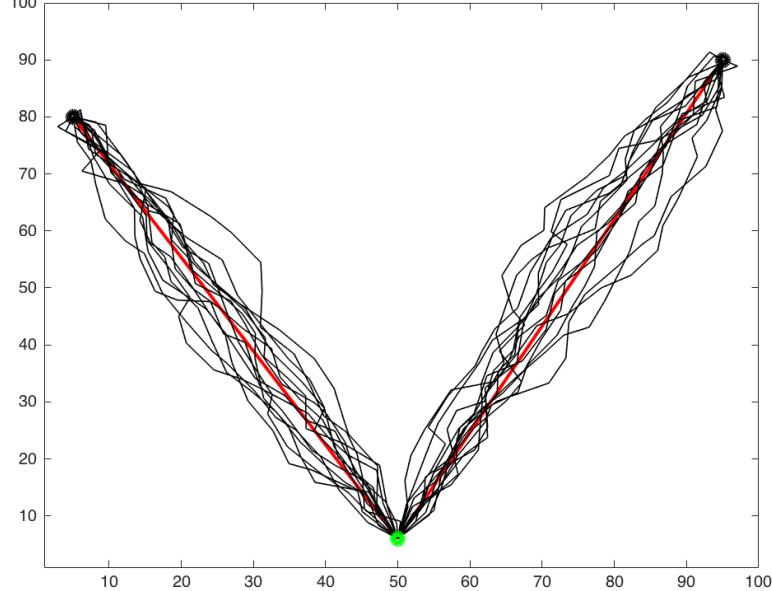


Figure 3.2: Monte Carlo paths (black) surrounding deterministic N-NHV paths (red). The starting point, $(\mathbf{s} = [50 \ 6]^T)$, is indicated in green. The set K_N contains $N = 2$ endpoints ($K_N = [5 \ 80]^T, [95 \ 90]^T$). $M_{mc} = 15$ random walks are generated for each endpoint. $\alpha = 5$. The variance of the Wiener process states are $\frac{1}{2}\alpha$.

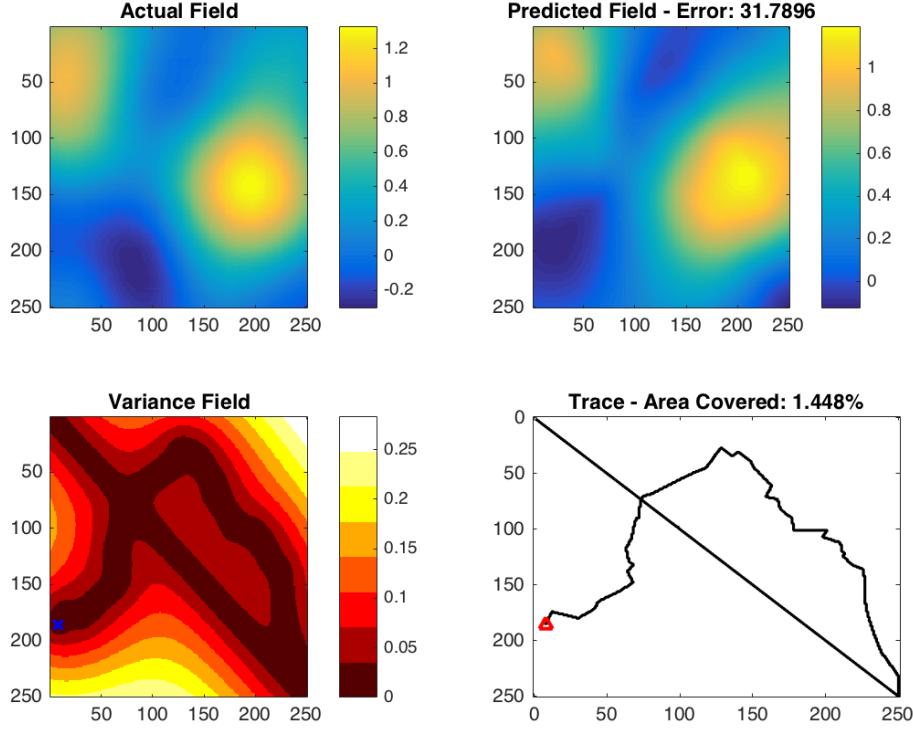


Figure 3.3: The Monte Carlo Path Planner (MCPP) algorithm (Section 3.4 terminated after two iterations of the algorithm. The actual field that is being explored is shown in the upper left. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle’s path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters. Here, N , the number of endpoints selected is $N = 3$, and the number of random paths per cord calculated is 15.

3.5 Discussion on the Planners Introduced

The Monte Carlo based path planner (MCPP) takes into account a set of noisy trajectories, and compares their estimated return on investment. The NHV simply takes the path to the highest prediction uncertainty location. Given enough trajectories (as N and M_{mc} get larger), the algorithm could find a path that will reduce overall field uncertainty in a

more brute-force way over the NHV algorithm. The disadvantages of The Monte Carlo Path Planner lay in the fact that the cost of each next move taken, are not considered directly. Rather, an entire trajectory is considered all at once. A more optimal approach to this planner would be to take into consideration the cost of each waypoint selected. Furthermore, the MCPP approach is the most expensive method introduced because of the need to re-predict the field for each candidate trajectory calculated. The NHV planner is the simplest to compute because it only requires one run of the Kriging field predictions and variance calculations, followed by a simple search for the highest field variances.

Chapter 4

Simulation Framework

Using *MATLAB*, the methods described in Chapter 3 on Path Planning were implemented to provide a simulation environment to show the effectiveness of the introduced methods. The target fields in the simulations are of a variable size with a single state of interest that can be sampled. The autocorrelation factor of the field in the simulation is an adjustable variable that determines the likeliness of each pair of neighboring points as a function of distance.

4.1 Simulated Exploration Vehicle Model Dynamics

The dynamics of the simulated vehicle are modeled after the vehicle model described in Equation 3.3. In the simulation, ΔT and V are constants predefined in the simulation environment. For a given waypoint destination, the vehicle will pass over every vesicle on the line connecting its original position and its final position. The vehicle's heading angle and speed can be controlled at any point in the simulation. The path planner directly feeds

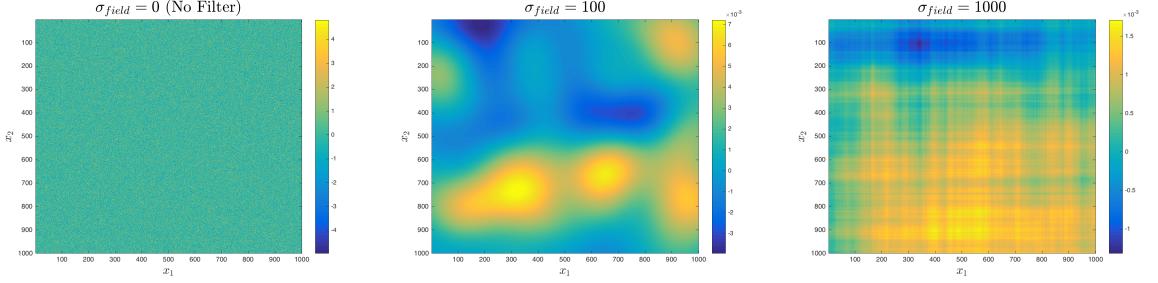
a control heading angle to the simulated vehicle. The speed of the vehicle is kept constant. A trajectory (a set of waypoints), T , calculated in Chapter 3, is loaded into a waypoint queue for the vehicle, where each waypoint is met one after another. After meeting the final waypoint in the trajectory, the next set of waypoints is calculated by a path planner. The process continues until the termination condition (maximum area scanned) is satisfied. A sample is taken at every possible vesicle that the vehicle passes over. The location and value of each sample is stored in the vehicle object's memory for later use in the prediction procedures.

4.2 Generating a Target Field

The simulation yields a target field that is of variable height h , and width w . Each vesicle in the field is exactly the area of the sensor footprint of the simulated vehicle's sensor. This is to make the sensor measurements as ideal as possible, so no samples are missed when a vesicle is flown over.

The field is composed of a single feature which is geospatially autocorrelated. Initially, the points on the field are generated from a normal distribution with a standard deviation of 1, and expected value of 0. The field is then convolved with a two dimensional Gaussian filter, of size $h \times w$, with a variable standard deviation, σ_{field} . The final filter "smooths" the field in order to simulate autocorrelation. The result is a randomly-generated, variably-sized, and autocorrelated field with a unit-less feature of interest. One such field can be observed in Figure 2.1a. As the value of the standard deviation of the Gaussian filter kernel, σ_{field} , increases, the field exhibits higher spatial autocorrelation. Inversely, when σ_{field} is

close to zero, the field becomes has no signs of spatial autocorrelation.



(a) $\sigma_{field} = 0$. The field exhibits no spatial autocorrelation.
(b) $\sigma_{field} = \frac{w}{10} = 100$. The field appears to exhibit some degree of spatial autocorrelation.
(c) $\sigma_{field} = w = 1000$. The field exhibits a high degree of spatial autocorrelation.

Figure 4.1: A field is generated using a zero mean normal distribution with variance 1. Varying degrees of spatial autocorrelation are shown for different values of σ_{field} .

4.3 Simulation Environment

The simulation, once started, runs a single exploration method at a time. A generated field is unknown to a vehicle object, but samples are collected as it passes along the field. The variances of Kriging predictions, the currently predicted field, and the path traversed are plotted along with the actual field being explored.

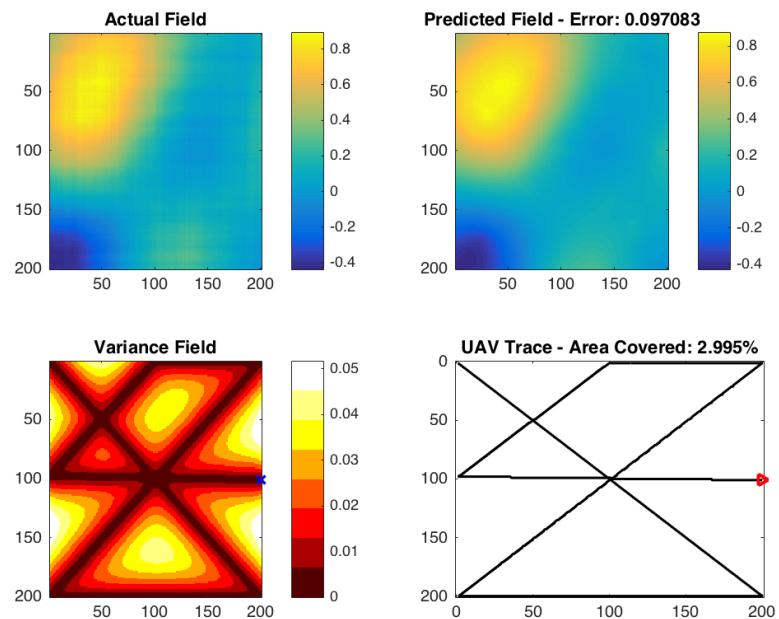


Figure 4.2: Next Highest Variance Path Planning. The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.

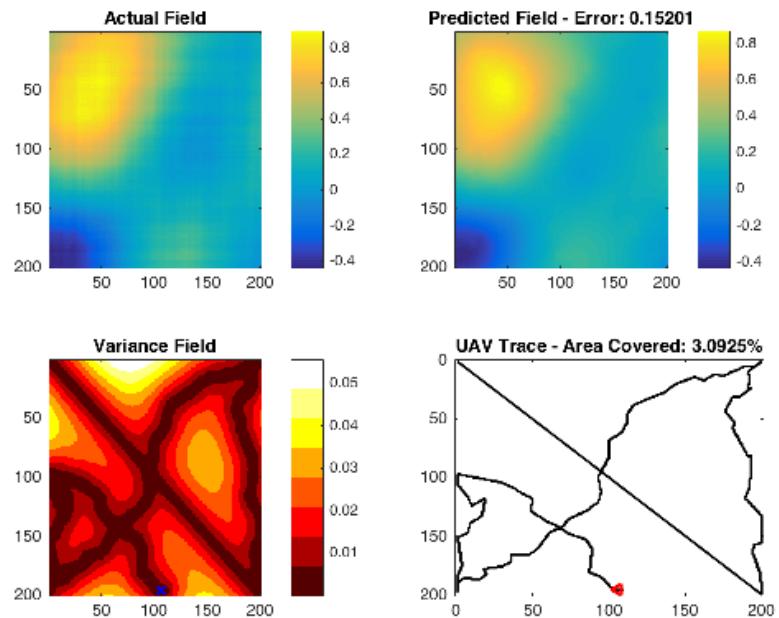


Figure 4.3: Monte Carlo path planner run ($N = 5$, $M_{mc} = 20$). The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.

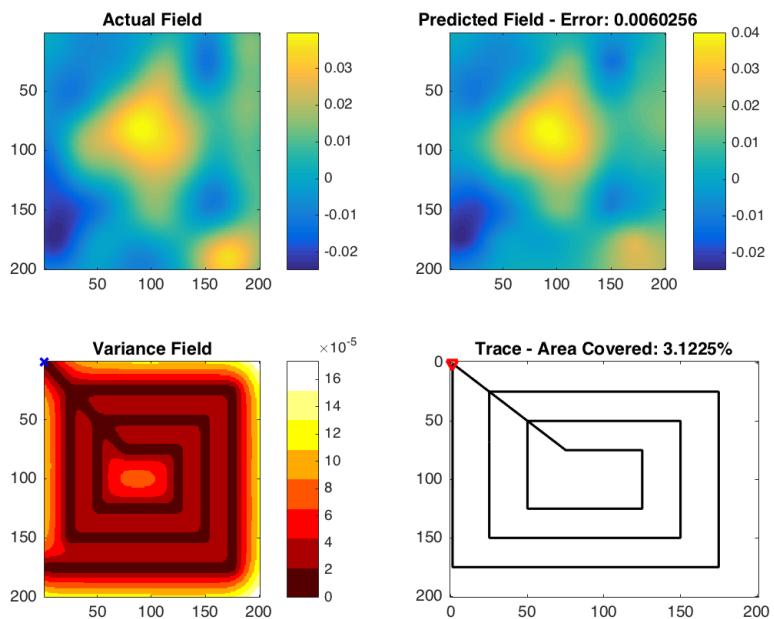


Figure 4.4: Zig-Zag method. The actual field (top left), predicted field (with error) (top right), variance field (bottom left), and traversed path (bottom right). The path planner is conducted on the same field with an approximate 3% scan area limit.

Chapter 5

Results

The three path planners (NHV, N-NHV, and MCPP) introduced in Chapter 3 all aim to reduce the overall prediction uncertainty of a field given a limited amount of flight time. They accomplish the task by calculating variances of a target field's predictions and attempting to choose a trajectory that reduces overall uncertainty.

5.1 Zig-Zag Path

A common approach to exploration and patrolling problems is the use of a zig-zag pattern. The methods introduced will be compared to a zig-zagging approach demonstrated in Nikhil Nigam, et al. *Control and Design of Multiple Unmanned Air Vehicles for a Persistent Surveillance Task* (Part II.C.3, Figure 6, [13]). The method will run a Kriging prediction and variance calculation on the samples taken using the zig-zag explorer, even though a prediction is not specified in the original work. This is to generate measurable and comparable metrics against the path planners introduced. The path planners introduced will be

compared against the zig-zag exploration method shown in Figure 5.1.

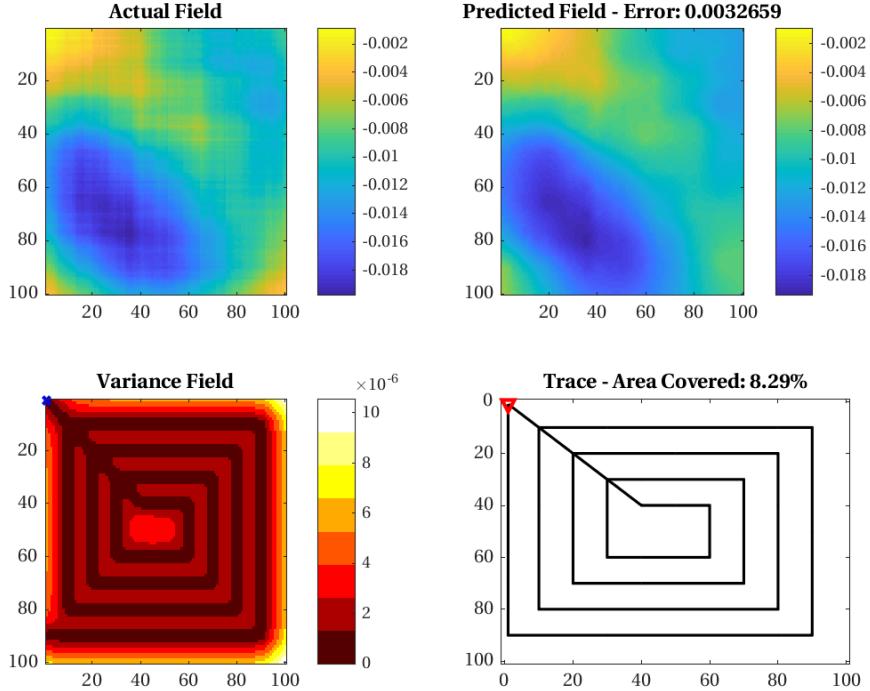


Figure 5.1: Zig-Zag exploration method set to scan a fixed percentage of a target field. A Kriging prediction and variance calculation is computed after completing the maneuver. The actual field that is being explored is shown in the upper left. The current prediction of the actual field is shown in the upper right. The variance of the current prediction of the field is shown in the lower left. The trace of the exploration vehicle’s path taken to the point of termination (in red) is shown in the lower right panel. All distance units are in meters.

5.1.1 Simulation Results

The methods introduced will be compared to one another and the zig-zag method for the same termination condition. Each method will stop the field exploration process when the exploration vehicle traverses a fixed path length expressed in terms of the area percentage scanned, A_{scan} . For example, if the maximum scan percentage of a size $w \times h$ size field is

$p\%$, then the method will stop exploring when $A_{scan} = \frac{p}{100}wh$ number of vesicles have been sampled. In an effort to allow the zig-zag method to cover as much of the field as possible, the spacing between each spiral bound, r , will be pre-calculated.

$$r = \frac{100}{p} \quad (5.1)$$

5.1.2 Simulation Result Parameters

The number of trajectories compared in both the NNHV and MCPP methods is $N = 5$. For the MCPP method, an additional $M_{mc} = 20$ Monte Carlo trajectories are calculated for each of the N trajectories. The target field size of the fields compared in the simulation have unit-less vesicle dimensions of 100×100 . Two random number generator seeds (2, 3) are used to generate two sets of runs in an effort to show the methods for a variety of random fields. The autocorrelation factors of the field will be varied in an effort to show the effectiveness of the methods for different field statistics. When the prediction variances of the methods are compared, the values are normalized to an a priori mean variance, which is equal to the mean variance of the field generated from running a Kriging prediction on the equivalent field from a set of samples taken from the first five forward diagonal vesicles on the target field.

5.1.3 Prediction Error Calculation

The quality of each path planner will be judged by its ability to explore a field in a fixed amount of time. The prediction error of each method will be used as a metric of path planning quality. The actual values of the fields scanned are known in simulation, and for each rerouting iteration, the predictions and prediction errors will be recalculated.

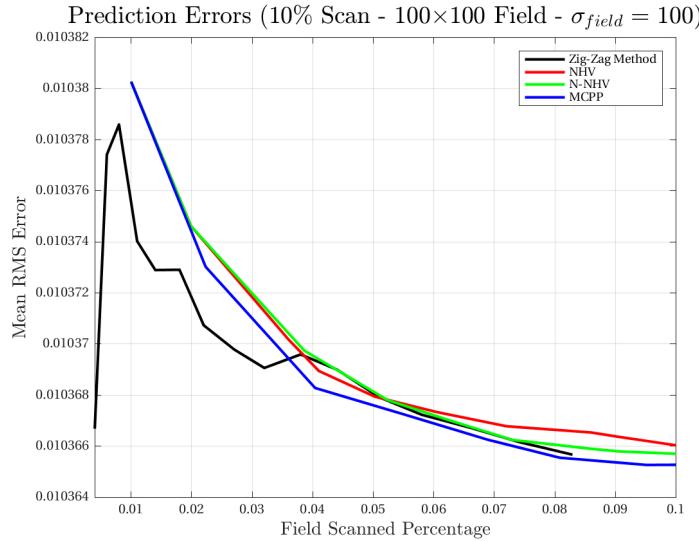
The prediction error function, $\text{erf}(Z, \hat{Z})$, will be the average root mean square (RMS) value for all N field predictions made, point by point, on the actual field, Z , and the predicted field, \hat{Z} .

$$\text{erf}(Z, \hat{Z}) = \frac{1}{N} \sum_{\forall i \in Z} (Z(\mathbf{s}_i) - \hat{Z}(\mathbf{s}_i))^2 \quad (5.2)$$

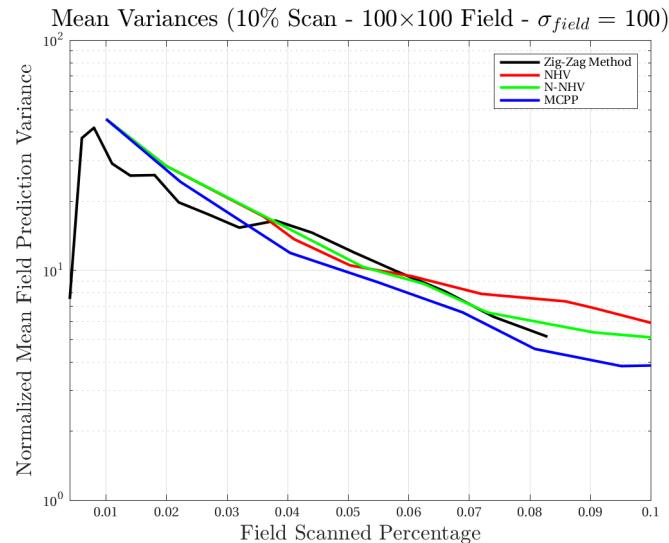
5.2 High Spatial Autocorrelation Results

The methods will be compared on target fields generated with the high degree of geospatial autocorrelation.

5.2.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.2: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.

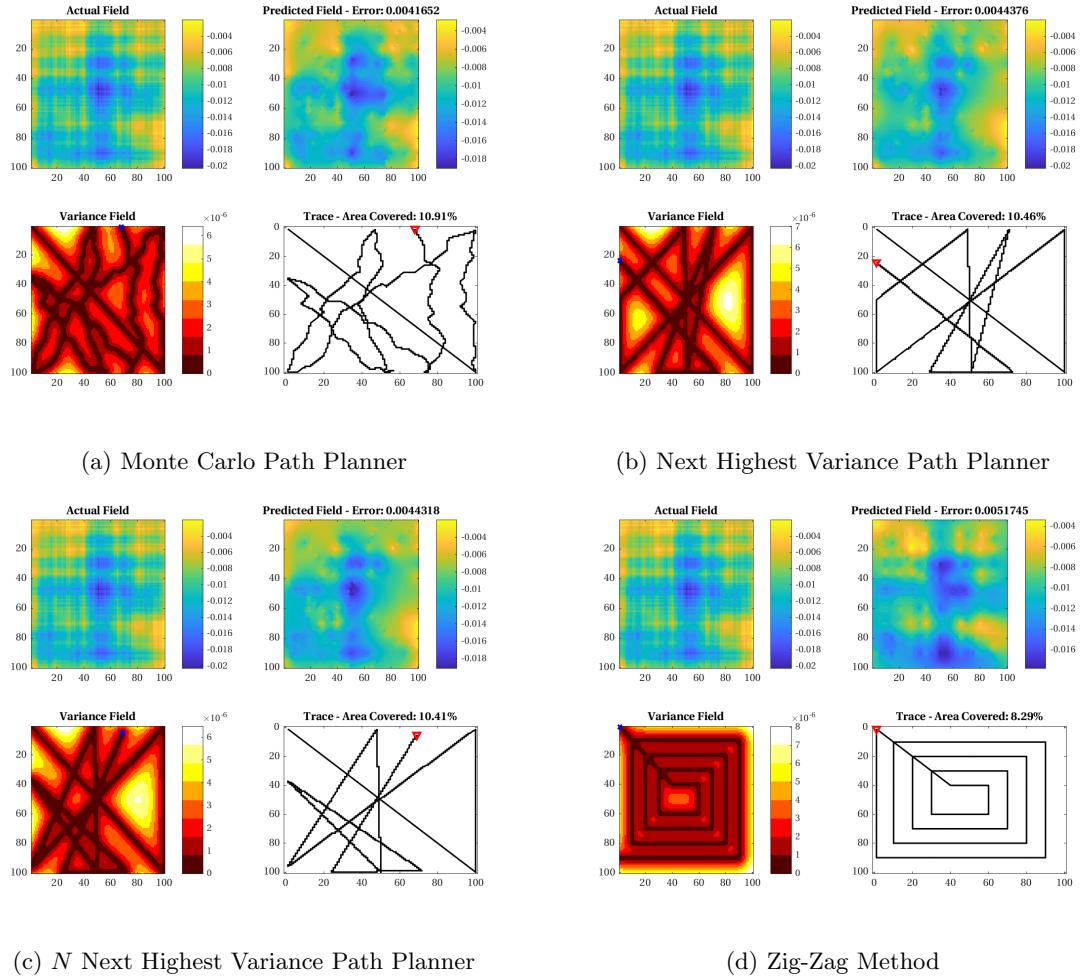
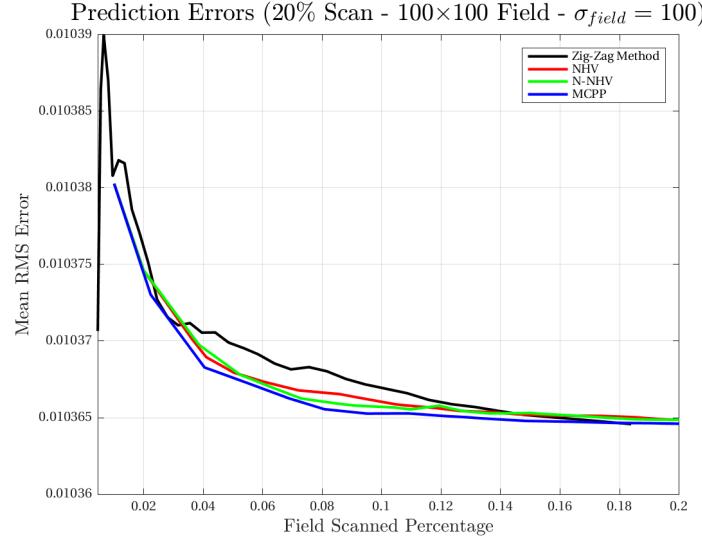


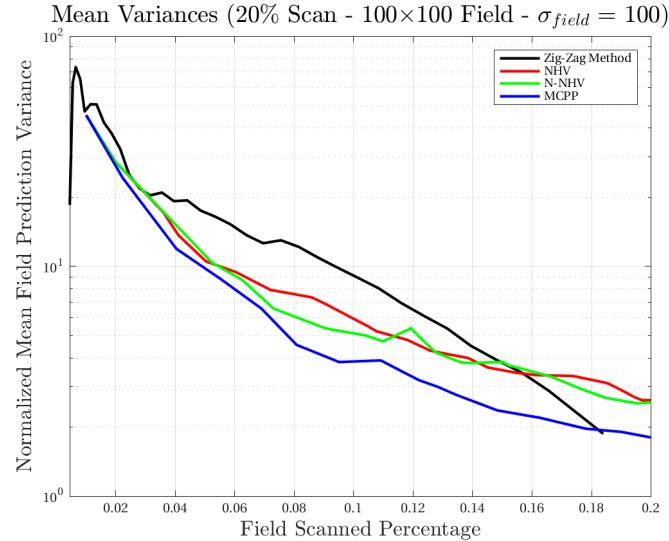
Figure 5.3: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 100$, random seed: 2.

5.2.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.4: A 20% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 100$, random seed: 2.

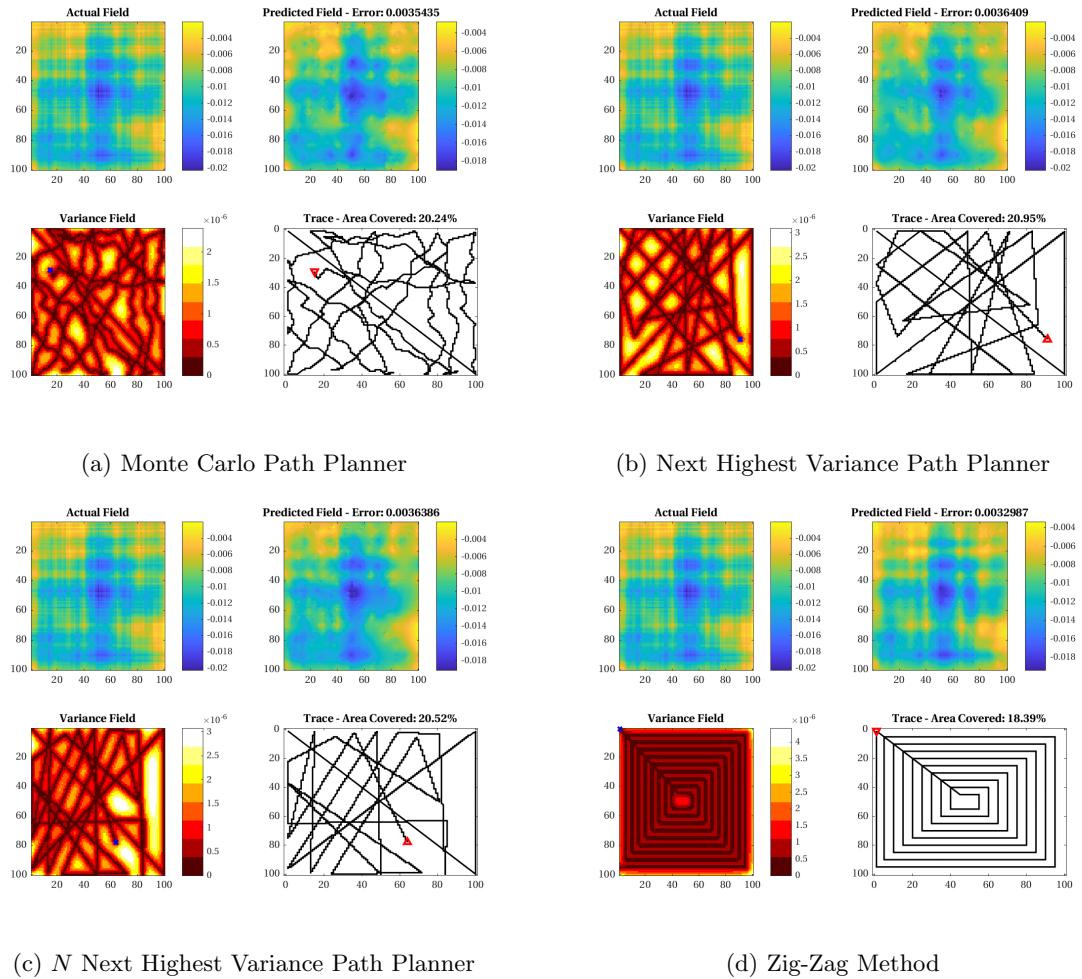
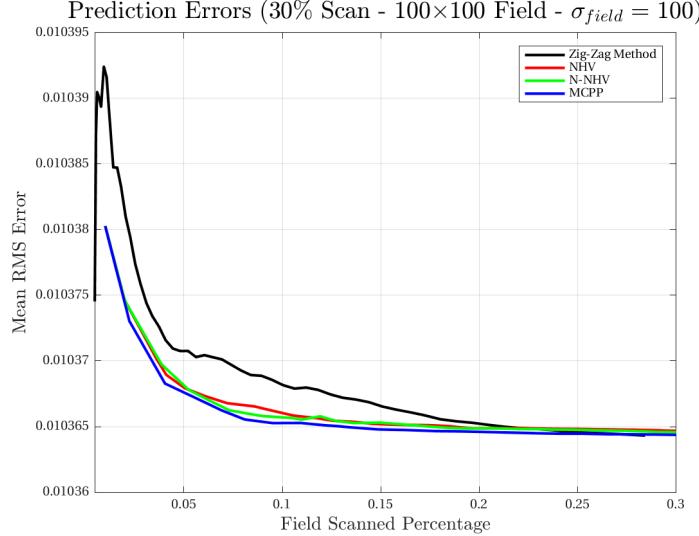


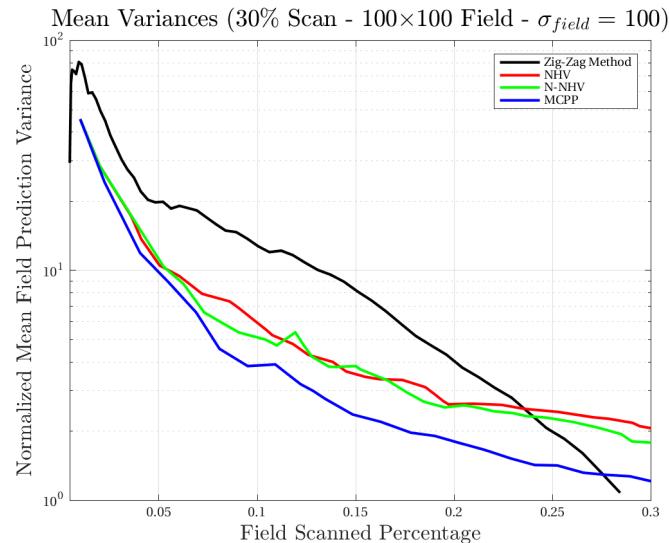
Figure 5.5: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 100$, random seed: 2.

5.2.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.6: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 2.

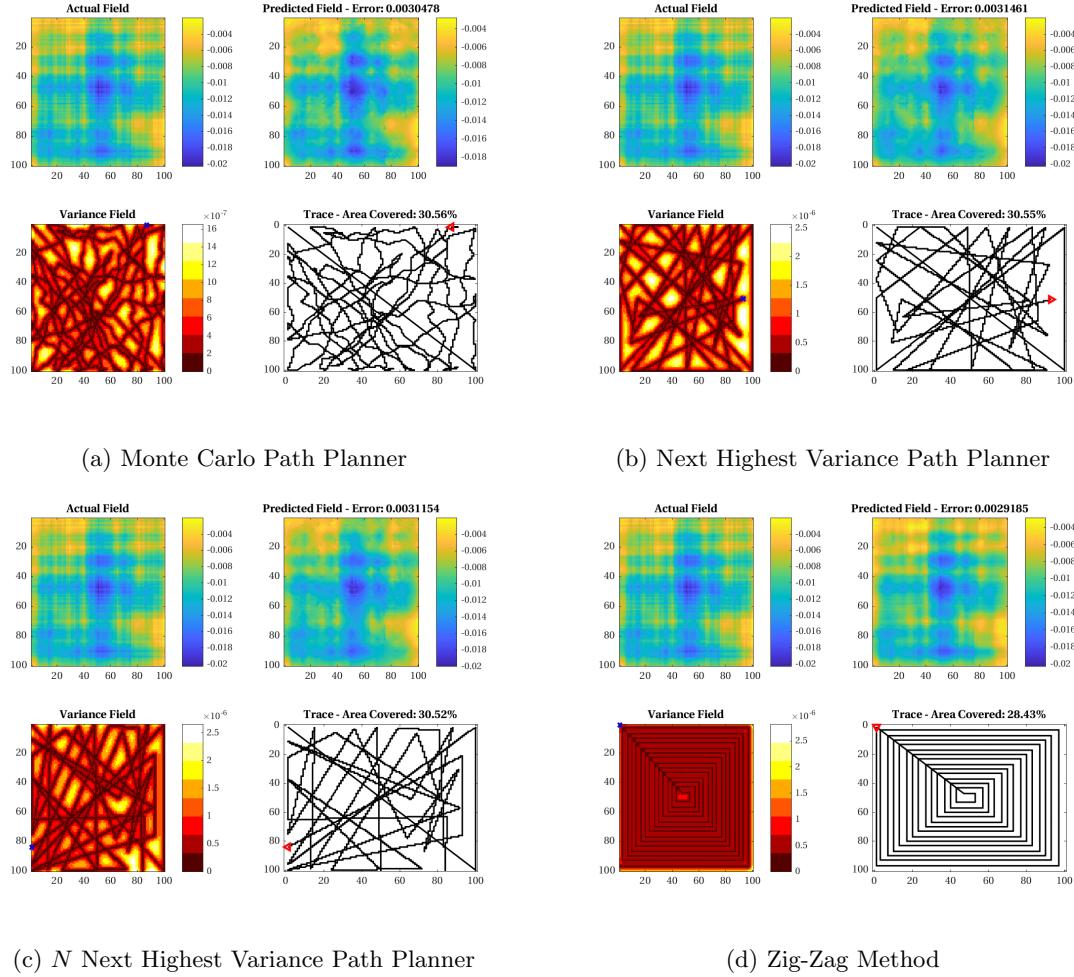


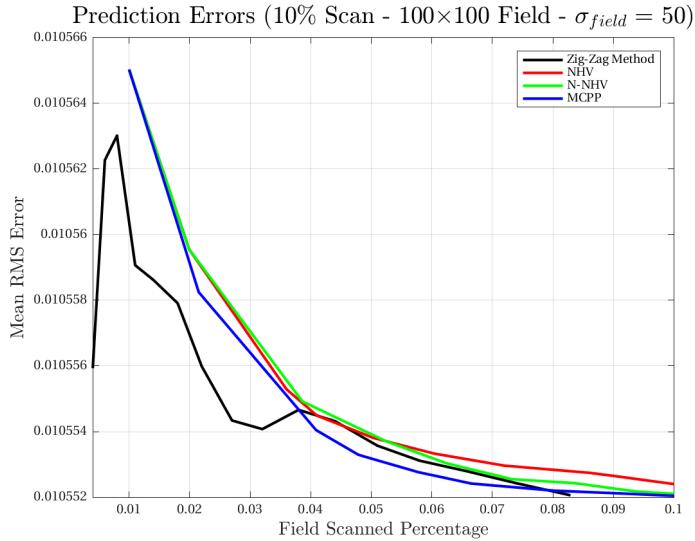
Figure 5.7: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 100$, random seed: 2.

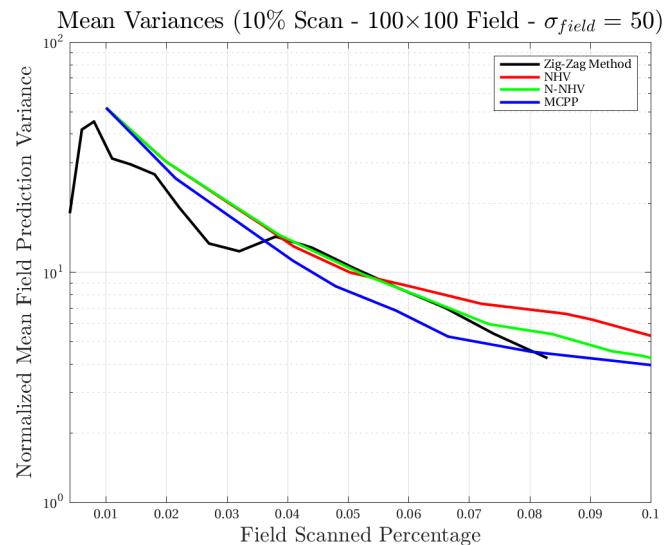
5.3 Half Width Spatial Autocorrelation Results

The methods will be compared on target fields generated with an autocorrelation factor that is half of the field width.

5.3.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.8: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.

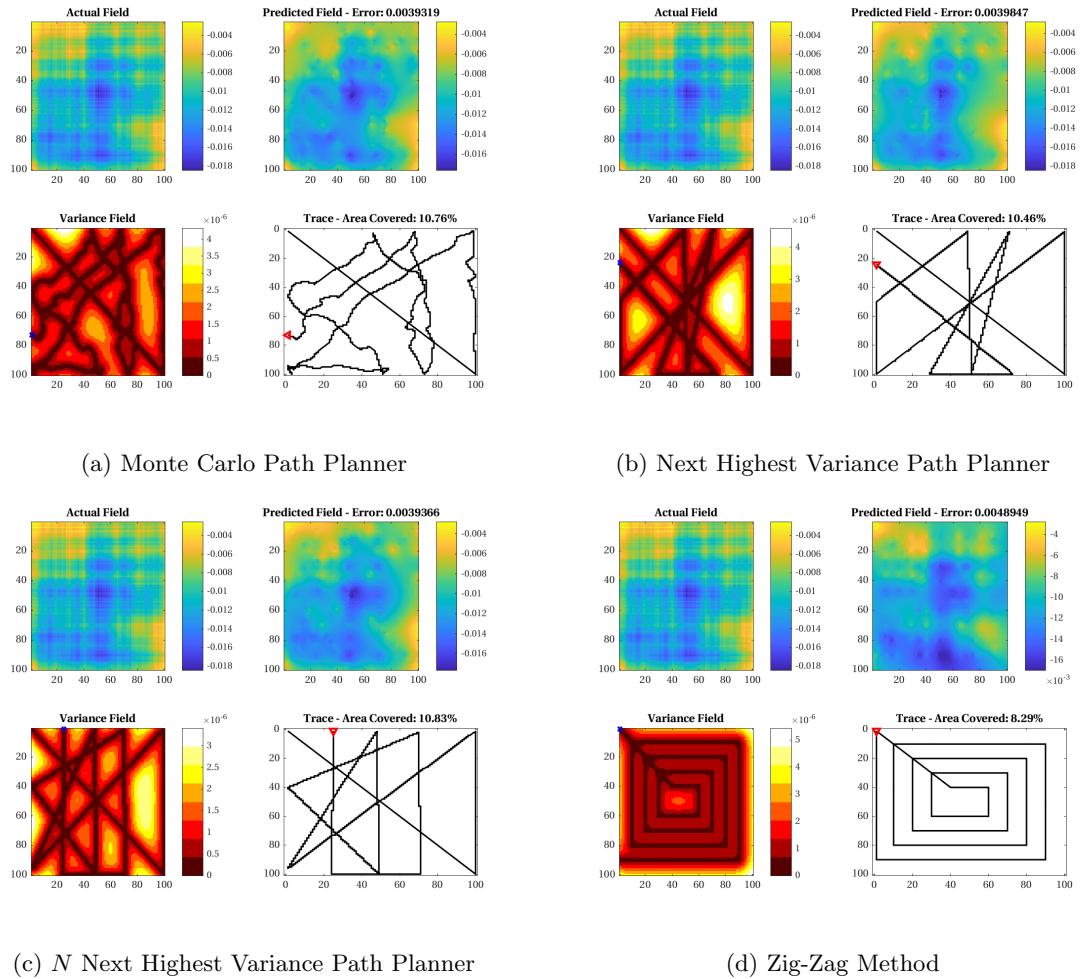
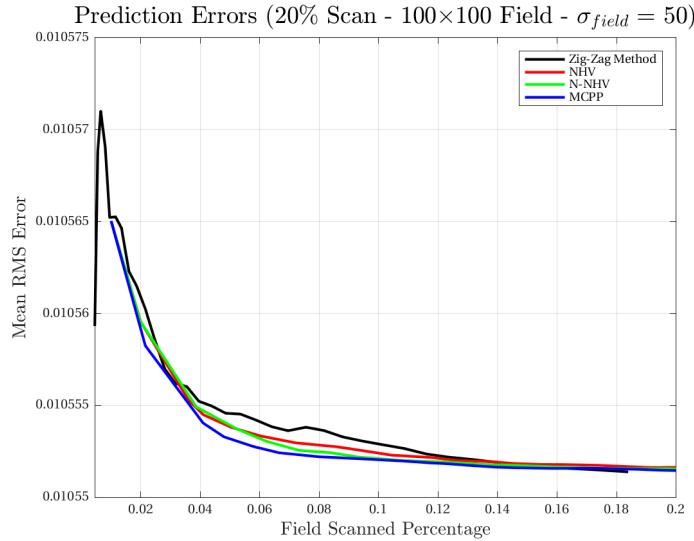


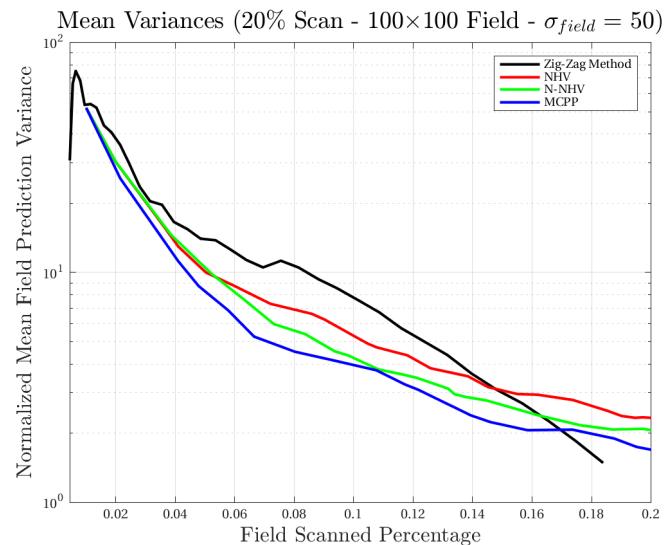
Figure 5.9: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 2.

5.3.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.10: A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.

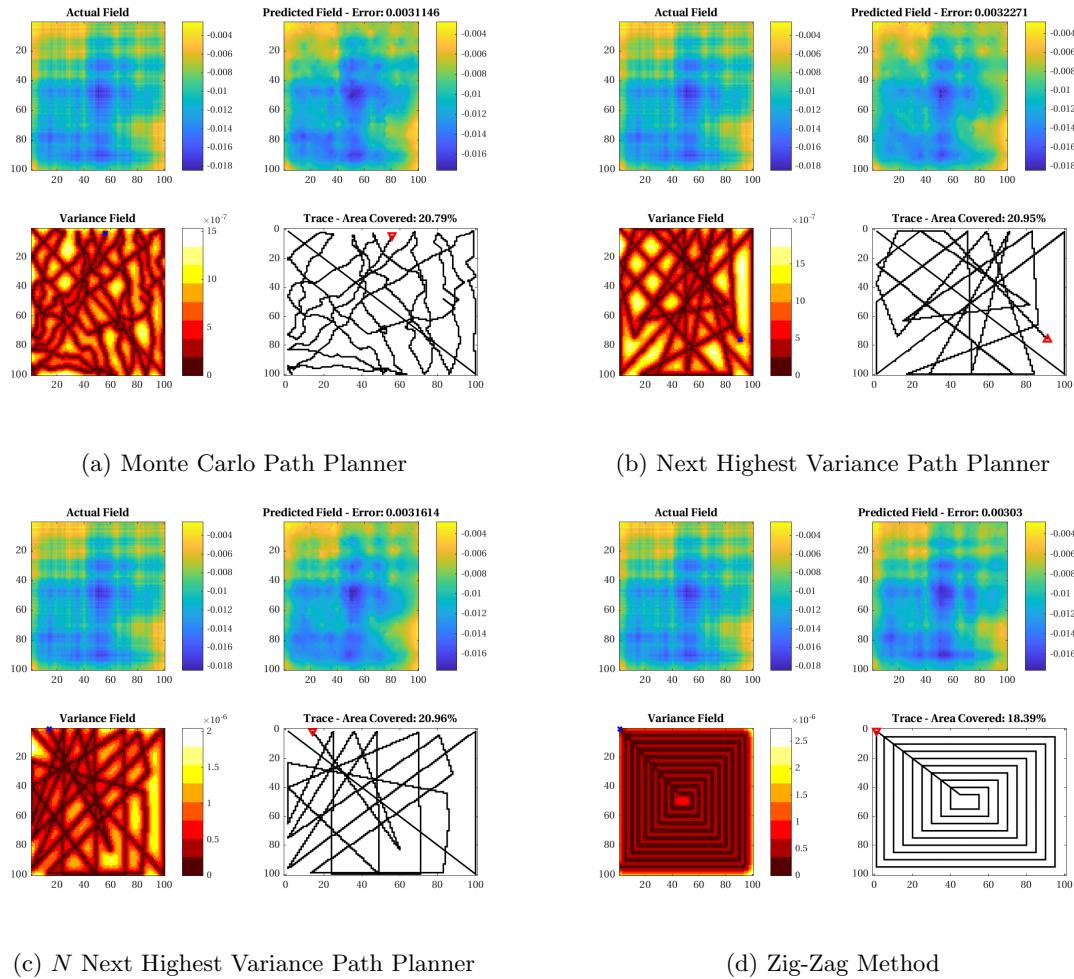
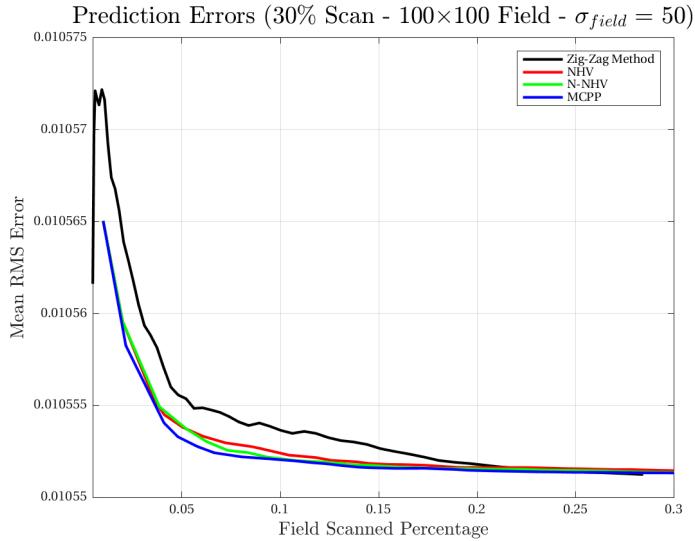


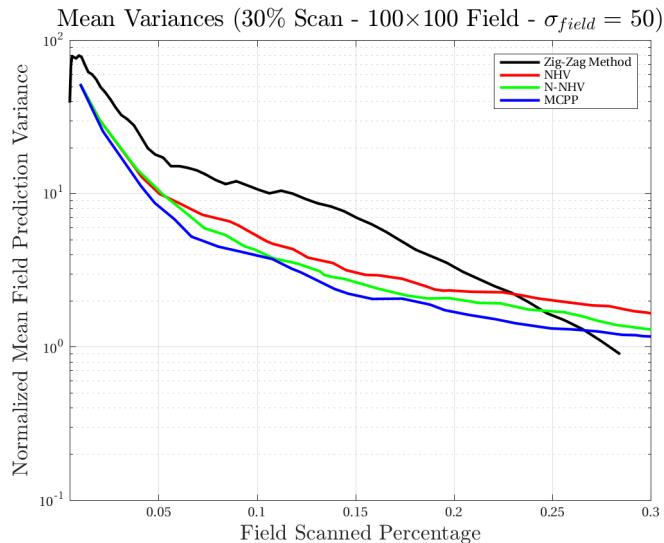
Figure 5.11: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 2.

5.3.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.12: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 2.

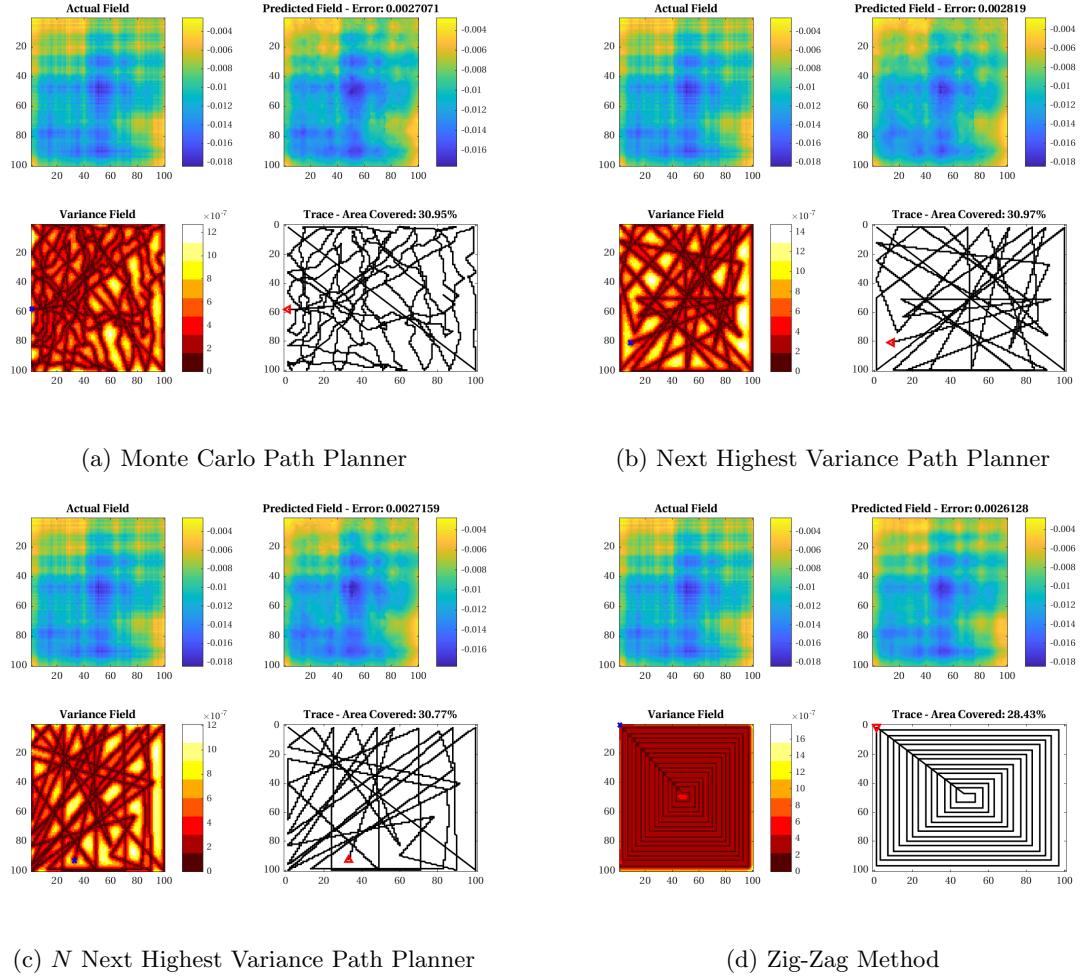


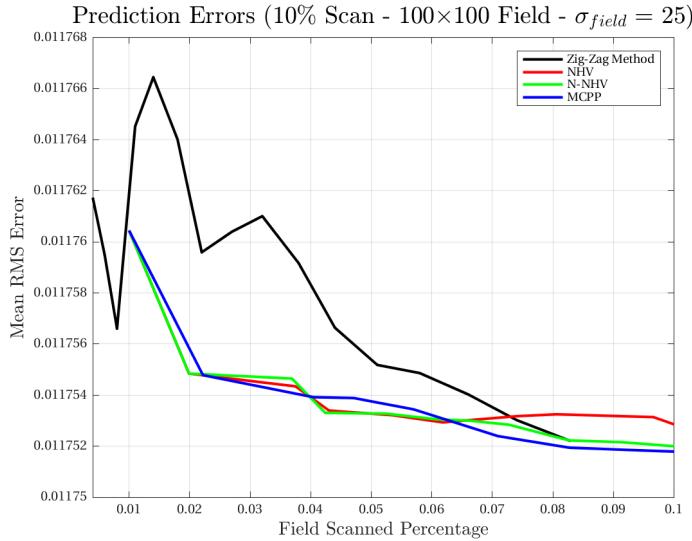
Figure 5.13: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 2.

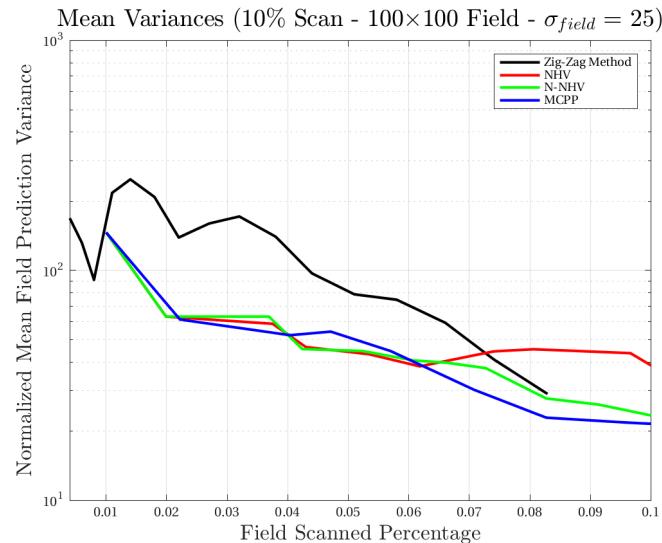
5.4 Quarter Width Spatial Autocorrelation Results

The methods will be compared on target fields generated with an autocorrelation factor that is one quarter of the field width.

5.4.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.14: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.

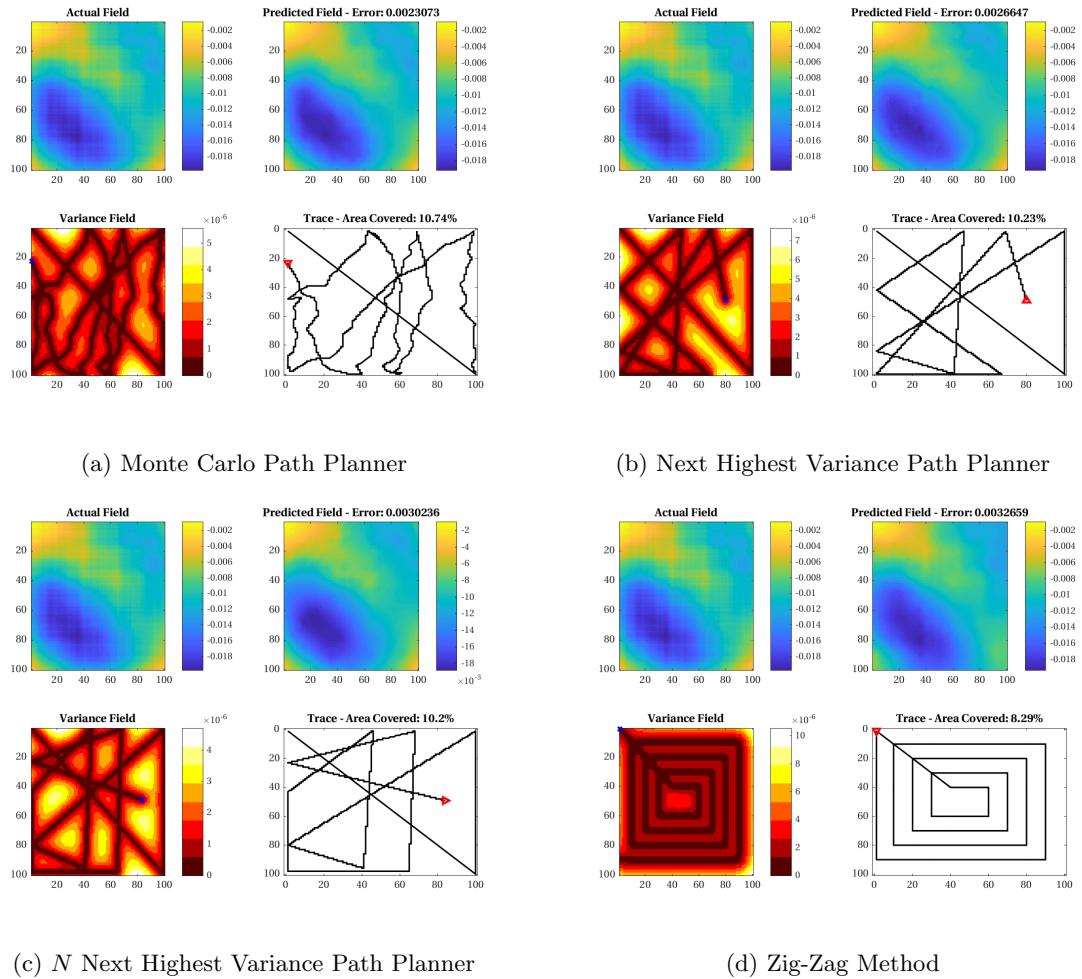
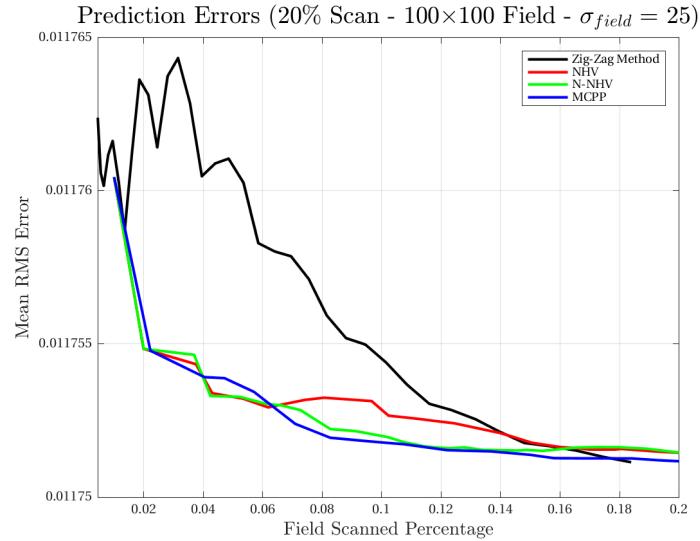


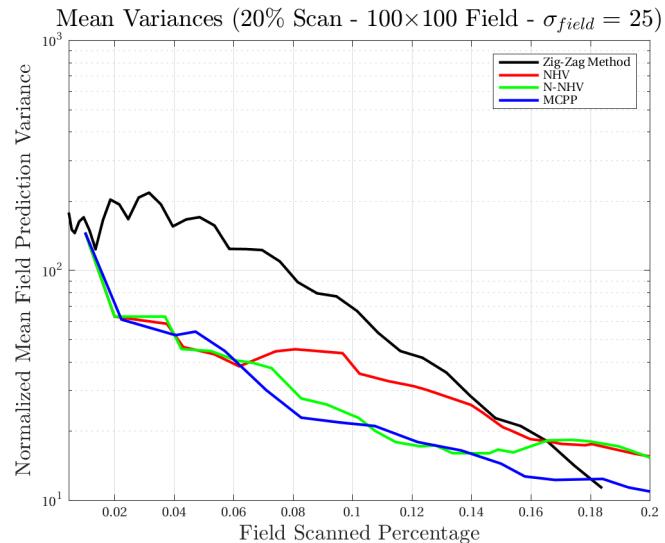
Figure 5.15: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 2.

5.4.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.16: A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.

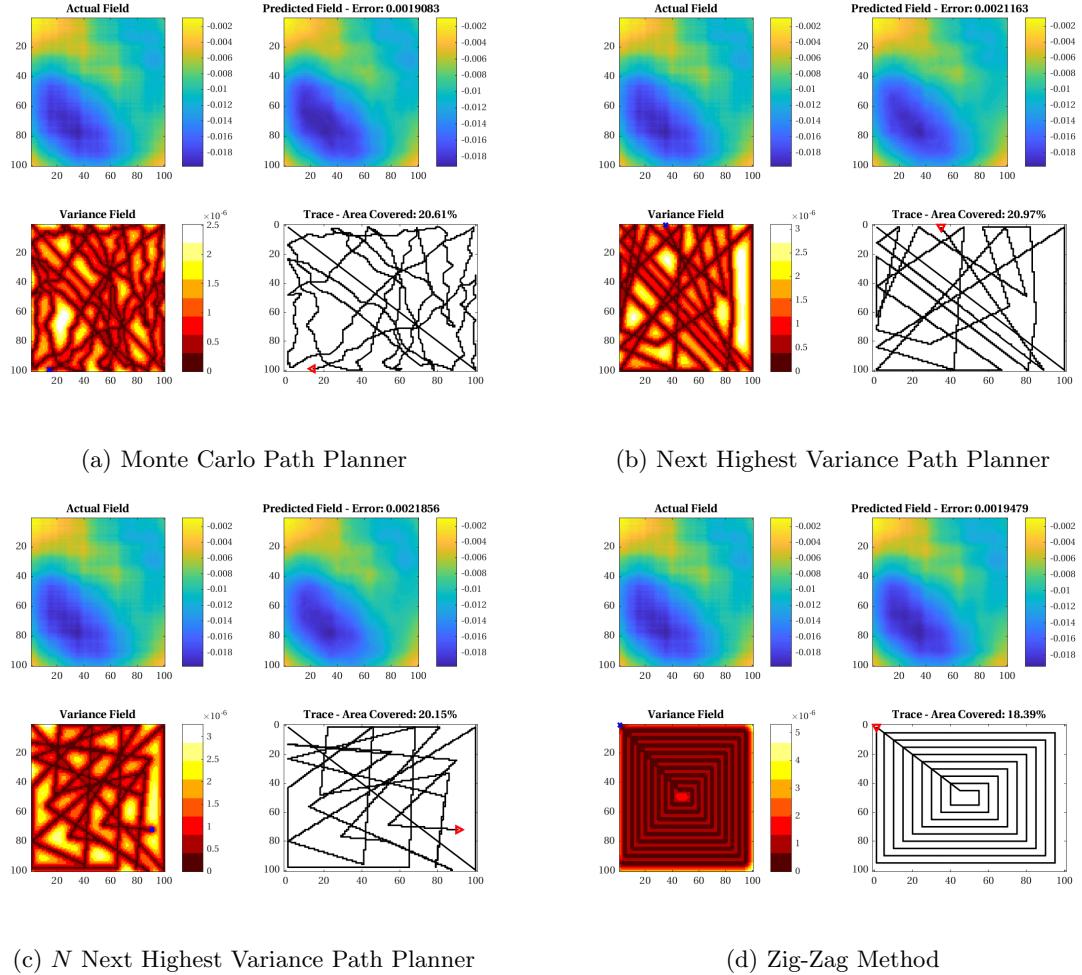
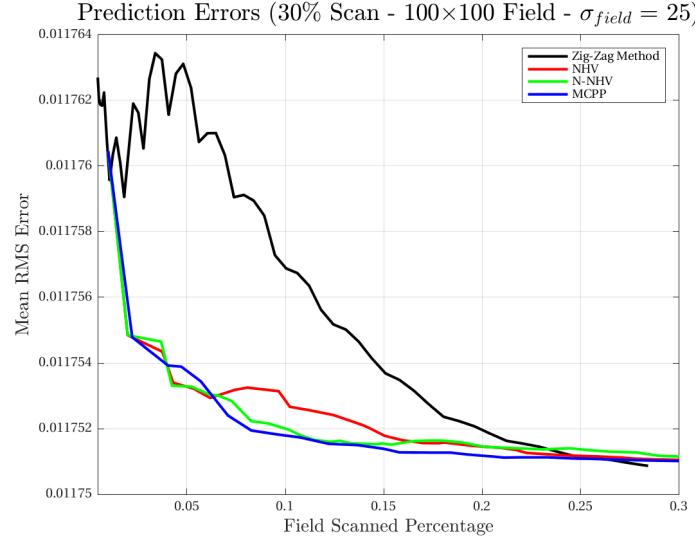


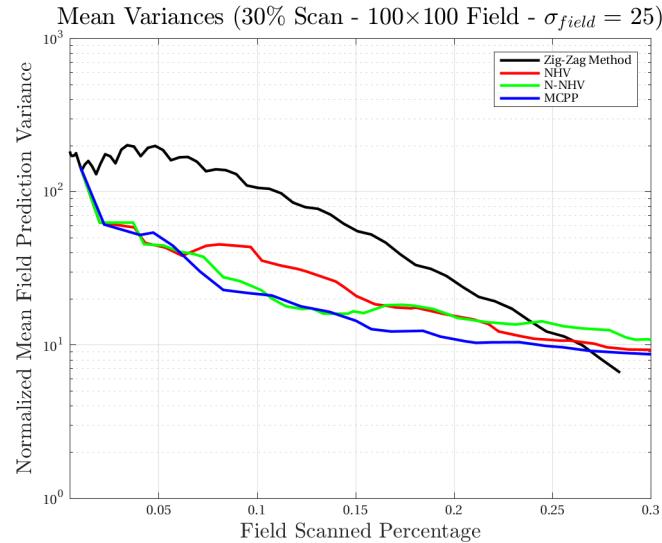
Figure 5.17: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 2.

5.4.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.18: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 2.

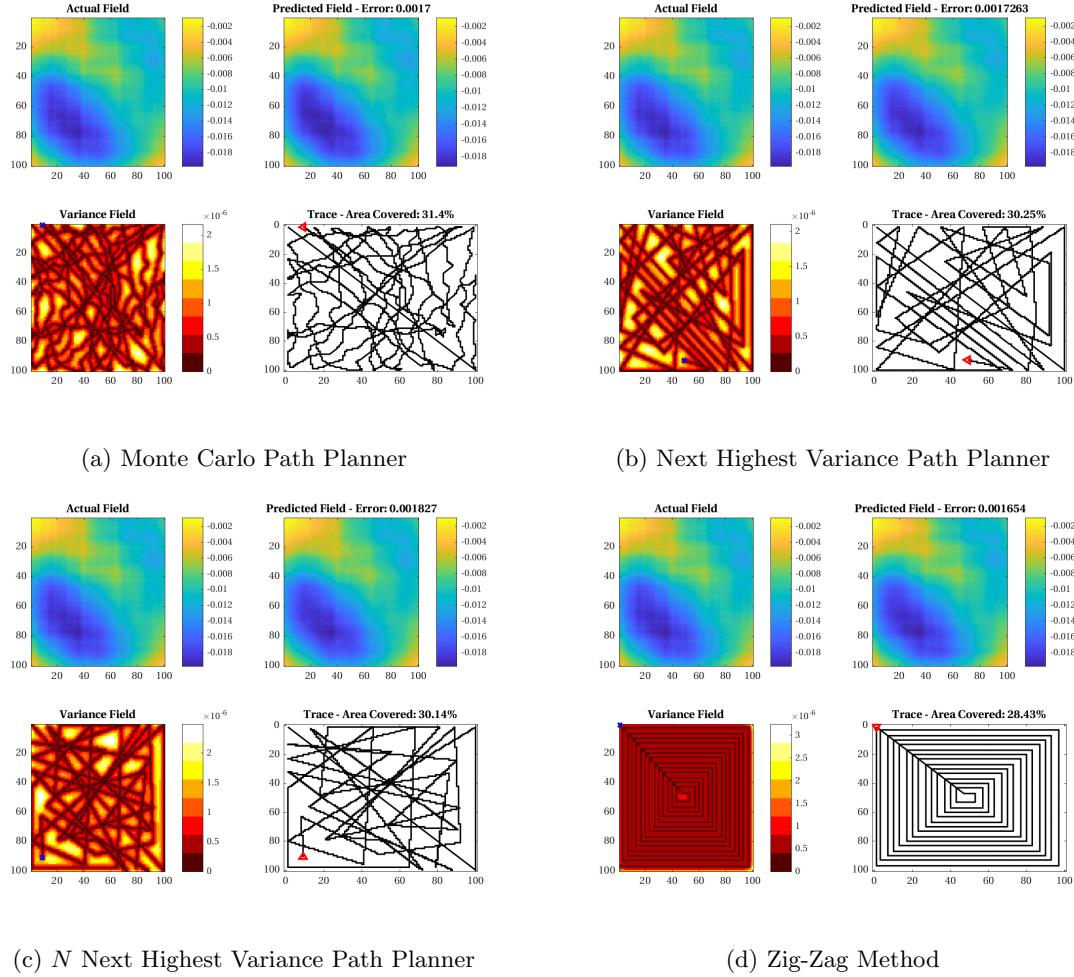


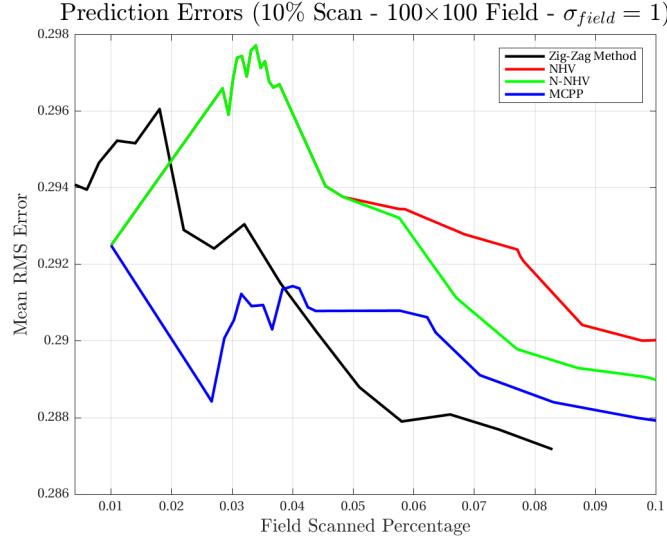
Figure 5.19: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 2.

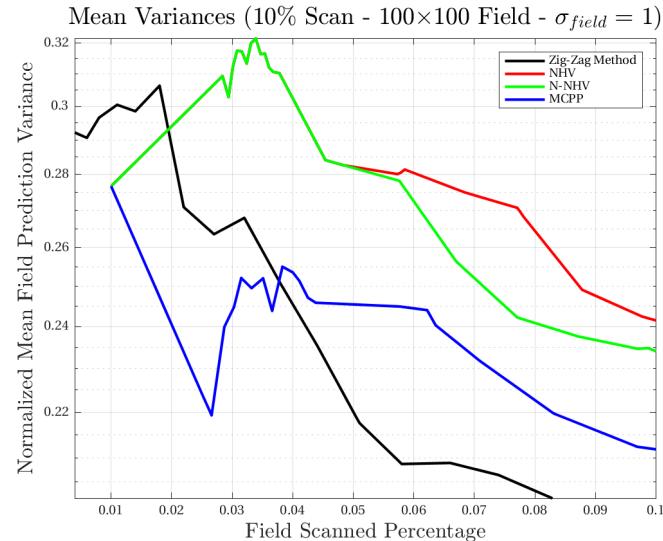
5.5 Low Spatial Autocorrelation Results

The methods will be compared on target fields generated with a low autocorrelation factor.

5.5.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.20: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 1$, random seed: 2.

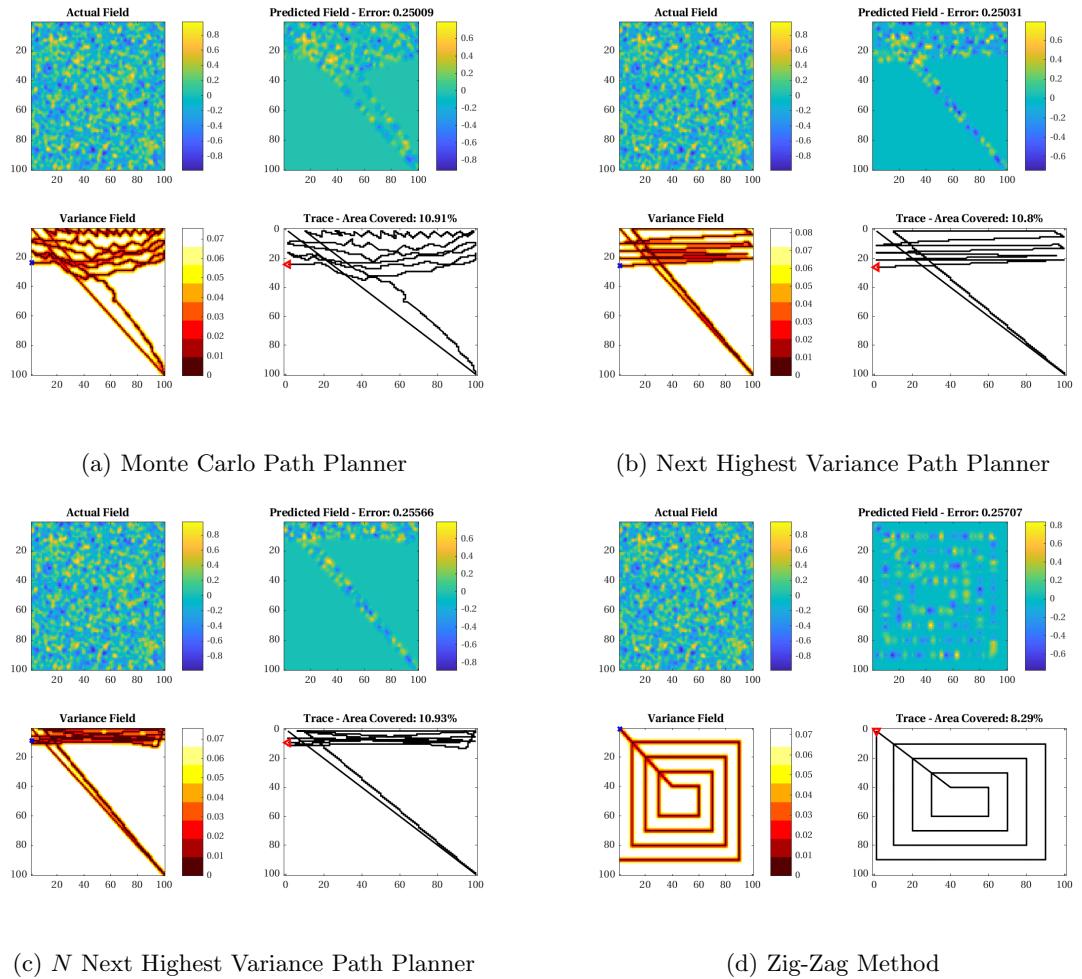
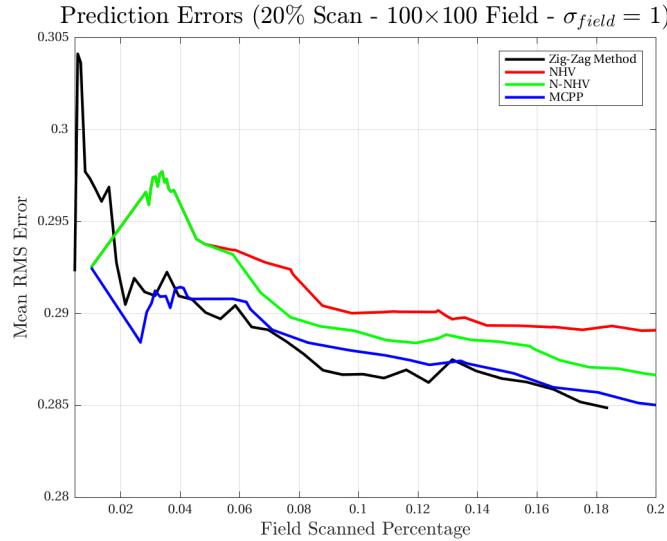


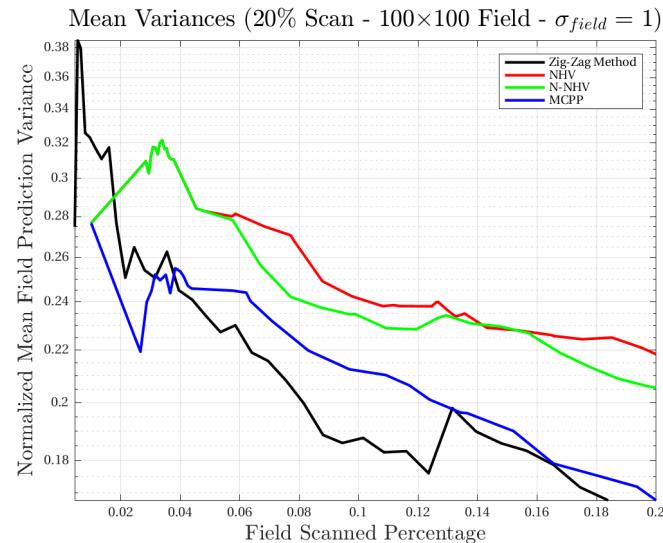
Figure 5.21: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 2.

5.5.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.22: A 20% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 1$, random seed: 2.

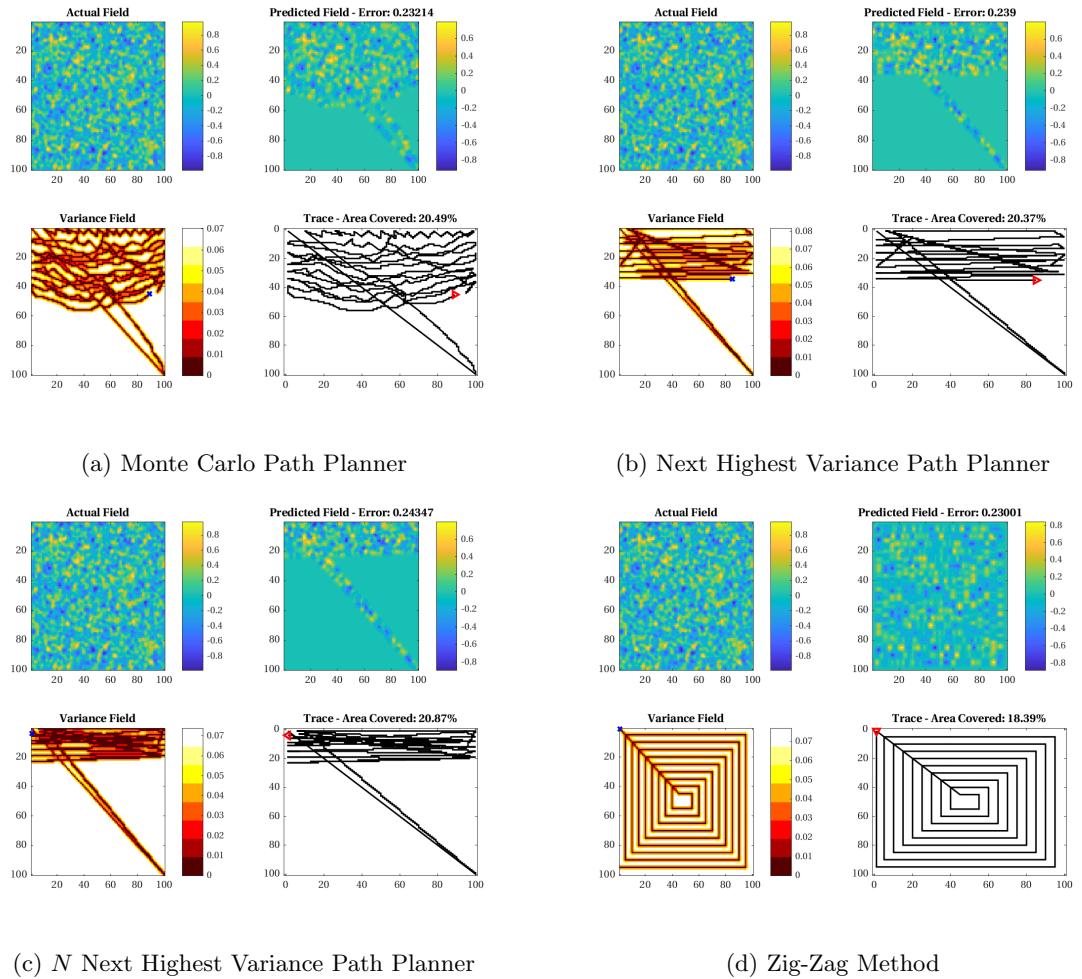
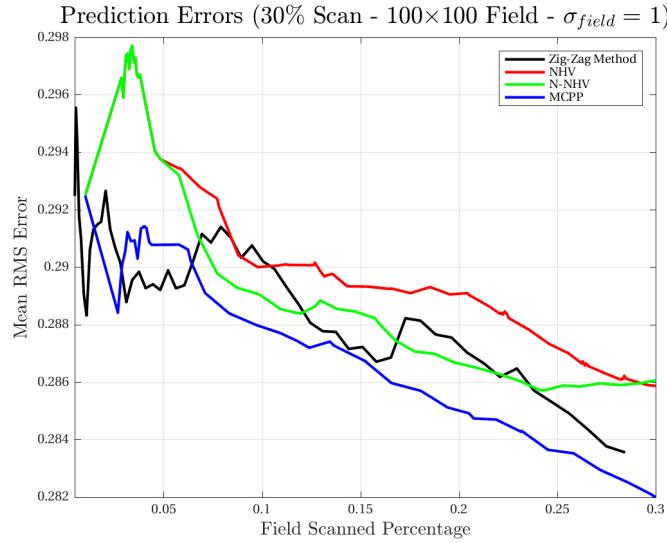


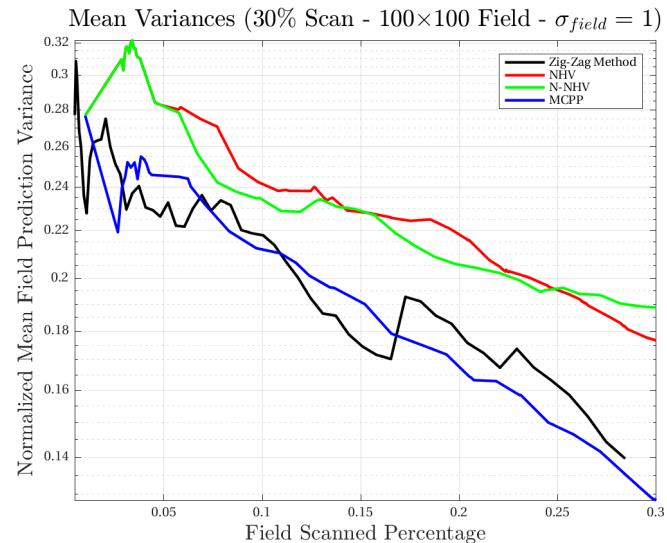
Figure 5.23: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 2.

5.5.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure 5.24: A 30% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 1$, random seed: 2.

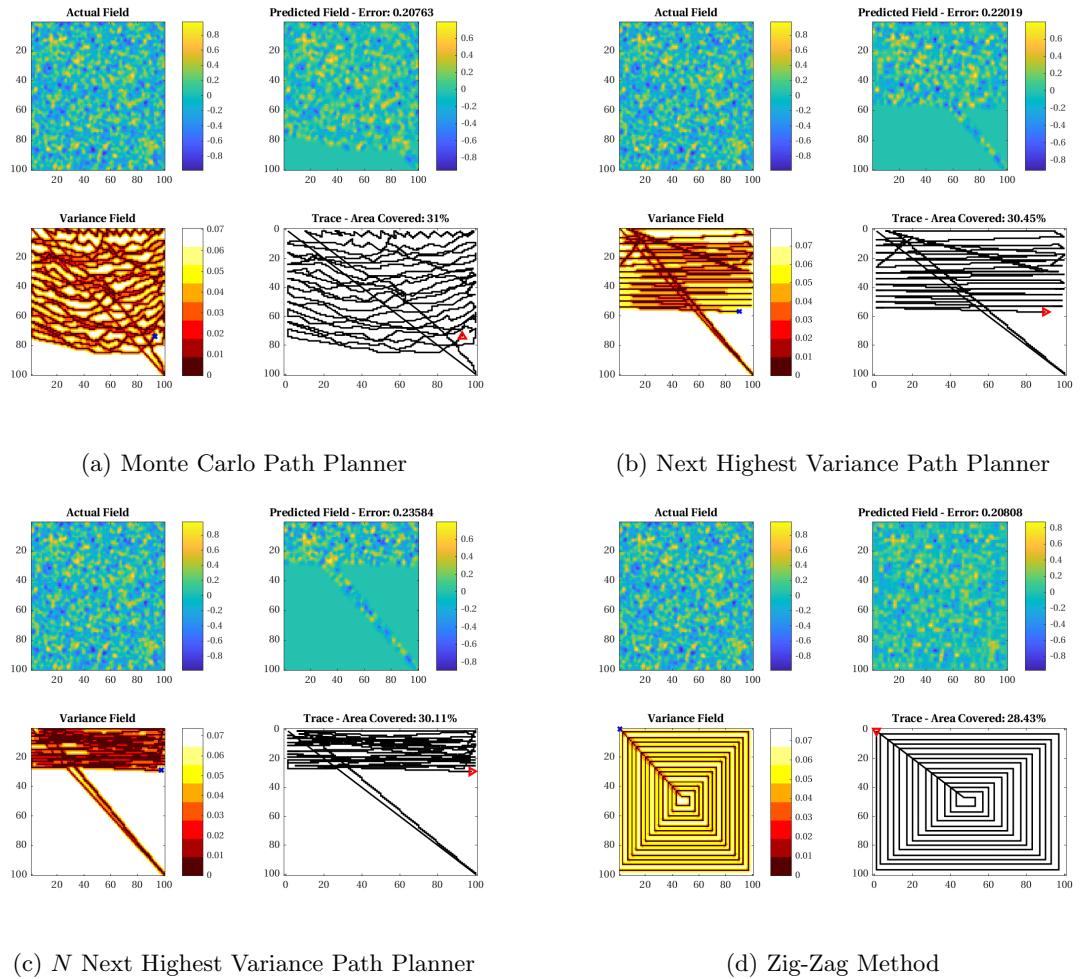


Figure 5.25: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 2.

Conclusion

The potential in a procedure using the Kriging Method as the core of a field exploration technique with an autonomous vehicle was demonstrated. By characterizing the confidence of the Kriging predictions made from observations in a field, along with uncertainty suppressing motivated path planners, the overall confidence in prediction of a target field as a whole can be maximized without having to scan every point. Furthermore, variance motivated path planning for prediction uncertainty suppression was shown to yield lower prediction errors and prediction variances for equal path lengths compared to predetermined paths.

An exploration vehicle could be maneuvered through a field to collect samples in areas of low Kriging prediction confidence. This in turn can increase the quality of prediction of the target field's state of interest to a higher degree of certainty. Using the Kriging variance generated path planning techniques, a more accurate prediction of a target field, though explored with the same path lengths as a fixed path spiral formation, was generated.

Future Work

Future works can be done in an effort to further develop Kriging prediction variance motivated path planning techniques. Firstly, a method which filters noisy measurements on a target field, with potentially dynamic properties, using a Kalman Filtering technique may be in the interest of a variance motivated path planner. The field states of a potentially dynamic field can be estimated using a Kalman Filter, as done in *Toward Dynamic Monitoring and Suppressing Uncertainty in Wildfire by Multiple Unmanned Air Vehicle System* by Sharon Rabinovich et al. [18]. The variances from both the Kalman filter state estimates, along with the Kriging prediction variances, could then potentially be fused in an effort to better track the states of a dynamic field by constructing paths along the field that suppress overall Kalman filter variances and Kriging prediction variances. This would ideally, in turn, produce predictions with higher certainty of state estimate for the overall field. The method could then also be used to help determine if a vehicle is suitable for tracking a certain state about the field. For example, if the dynamics of the field are found to be too dynamic for a simulated vehicle with a fixed velocity to track, while minimizing Kriging prediction variances and Kalman state variances, a more suitable vehicle might be chosen instead for that field. Furthermore, the Kriging state predictions and prediction

variances could serve as the initial prediction and variance values for the Kalman Filter.

Further work could also be the development of a comparison of the introduced methods, for varying vehicle dynamics, e.g. a Dubins Vehicle.

Bibliography

- [1] A. Davari R. Chen C. C. Castello, J. Fan. Optimal sensor placement strategy for environmental monitoring using wireless sensor networks. In *2010 42nd Southeastern Symposium on System Theory (SSST)*, pages 275–279, March 2010.
- [2] Howie Choset. Coverage for robotics – a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31(1):113–126, Oct 2001.
- [3] Noel A. C. Cressie. *Geostatistics*, pages 27–104. John Wiley & Sons, Inc., 2015.
- [4] C.V. Deutsch and A.G. Journel 2nd Edition. *GSLIB Geostatistical Software Library and User's Guide*. Oxford University Press, Oxford University Press, Inc. New York, NY, 1998.
- [5] Paulo A. F. Rezeck Guilherme A. Potje Luiz C. C. Benyosef Andr Wiermann Gustavo M. Freitas Luis G. U. Garcia Douglas G. Macharet, Hctor I. A. Perez-Imaz. Autonomous aeromagnetic surveys using a fluxgate magnetometer. *The International Journal of Robotics Research, Sensors* (Basel, Switzerland)(101204366), 2016.
- [6] Y.A. Felus, Alan Saalfeld, and B Schaffrin. Delaunay triangulation structured kriging for surface interpolation. 65:27–36, 03 2005.
- [7] Steven Fortune. *Handbook of Discrete and Computational Geometry*. CRC Press, Inc., Boca Raton, FL, USA, 1997.
- [8] Daniel Ozick Gregg W. Landry, David A. Cohen. Debris sensor for cleaning apparatus, 2004.
- [9] The MathWorks Inc. fminsearch reference. <https://www.mathworks.com/help/matlab/ref/fminsearch.html#References>. Accessed: 2018-03-30.
- [10] Lucas Janson, Edward Schmerling, and Marco Pavone. *Monte Carlo Motion Planning for Robot Trajectory Optimization Under Uncertainty.*, pages 343–361. Springer International Publishing, Cham, 2018.
- [11] Georges Matheron. Principles of geostatistics. *Economic Geology*, 58(8):1246–1266, 1963.

- [12] Harvey J. Miller. Tobler's first law and spatial analysis. *Annals of the Association of American Geographers*, 94(2):284–289, 2004.
- [13] N. Nigam. Control and design of multiple unmanned air vehicles for persistent surveillance. 2009.
- [14] Nikhil Nigam. The multiple unmanned air vehicle persistent surveillance problem: A review. *Machines*, 2(1):13–72, 2014.
- [15] J. Pulido Fentanes, A. Badiee, T. Duckett, J. Evans, S. Pearson, and G. Cielniak. Kriging-Based Robotic Exploration for Soil Moisture Mapping Using a Cosmic-Ray Sensor. *ArXiv e-prints*, November 2018.
- [16] S. Rabinovich. Multi-uav path coordination based on uncertainty estimation. In *UC Santa Cruz: Division of Graduate Studies*, 2017.
- [17] S. Rabinovich, R. E. Curry, and G. H. Elkaim. A methodology for estimation of ground phenomena propagation. In *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 1239–1244, April 2018.
- [18] S. Rabinovich, R. E. Curry, and G. H. Elkaim. Toward dynamic monitoring and suppressing uncertainty in wildfire by multiple unmanned air vehicle system. *The International Journal of Robotics Research*, 2018.
- [19] Eduard Semsch, Michal Jakob, Dušan Pavlíček, and Michal Pěchouček. Occlusion-aware multi-uav surveillance. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1 - Volume 1*, AAMAS '10, pages 1407–1408, Richland, SC, 2010. International Foundation for Autonomous Agents and Multiagent Systems.
- [20] Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, pages 517–524, New York, NY, USA, 1968. ACM.
- [21] Andy Swan. Goovaerts, p. 1997. geostatistics for natural resources evaluation. applied geostatistics series. xiv 483 pp. new york, oxford: Oxford university press. price 46.95 (hard covers). isbn 0 19 511538 4. *Geological Magazine*, 135(6):819842, 1998.
- [22] W. R. Tobler. A computer movie simulating urban growth in the detroit region. *Economic Geography*, 46:234–240, 1970.
- [23] S.C. Van der Graaf. Natural neighbour kriging and its potential for quality mapping and grid design. *TU Delft Repository*, 2016.

- [24] Cheng Zhang and Hailong Pei. Oil spills boundary tracking using universal kriging and model predictive control by uav. In *Proceeding of the 11th World Congress on Intelligent Control and Automation*, pages 633–638, June 2014.

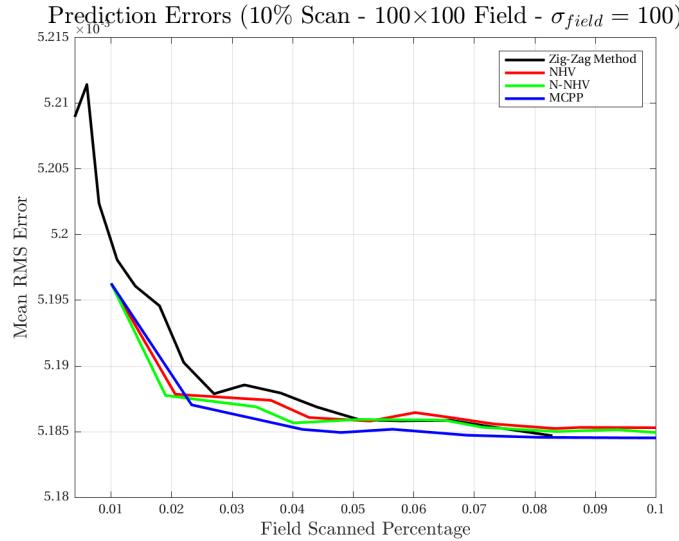
Appendix A

Additional Results

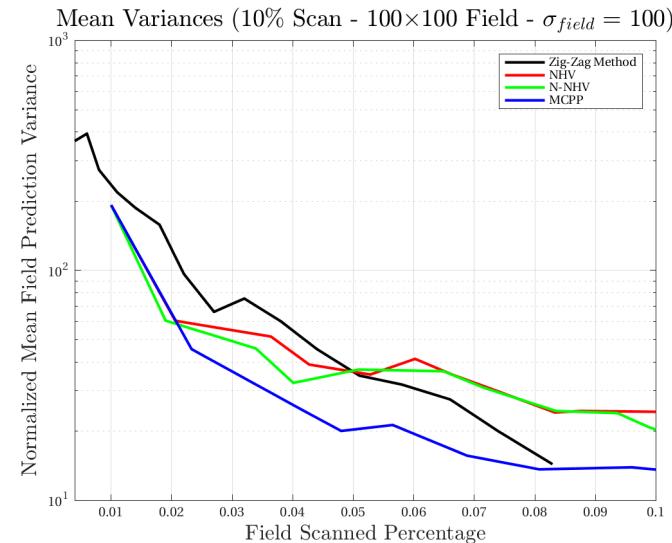
A.1 High Spatial Autocorrelation Results

The methods will be compared on target fields generated with the high degree of geospatial autocorrelation.

A.1.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.1: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.

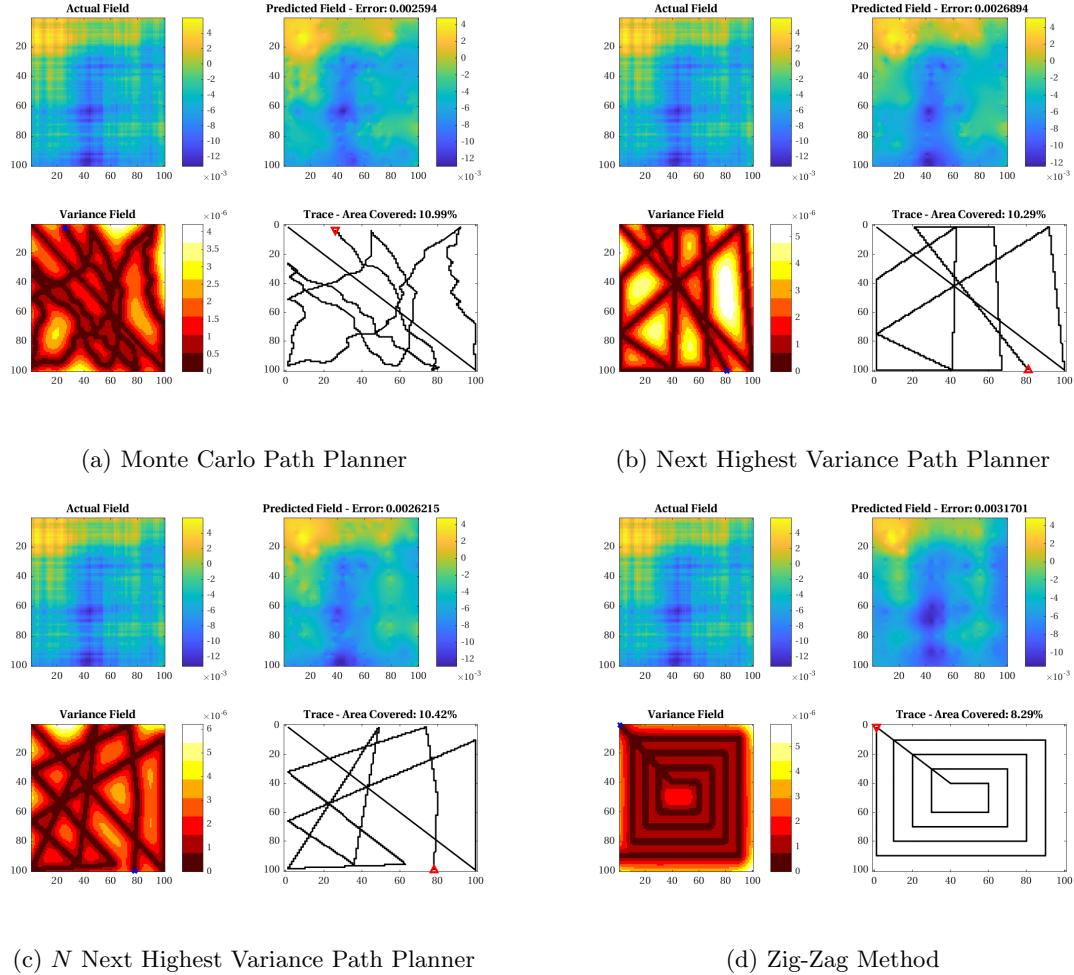
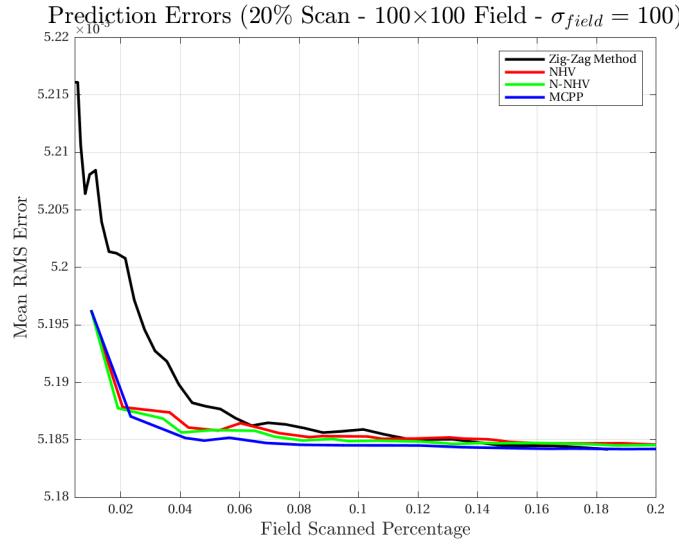


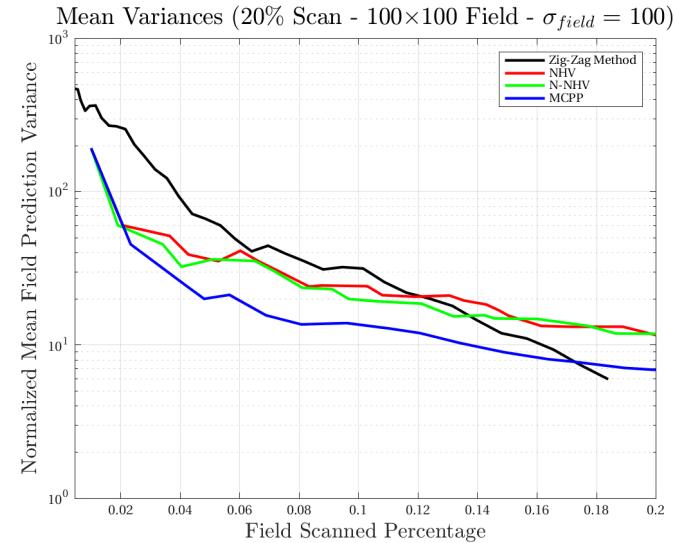
Figure A.2: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 100$, random seed: 3.

A.1.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.3: A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.

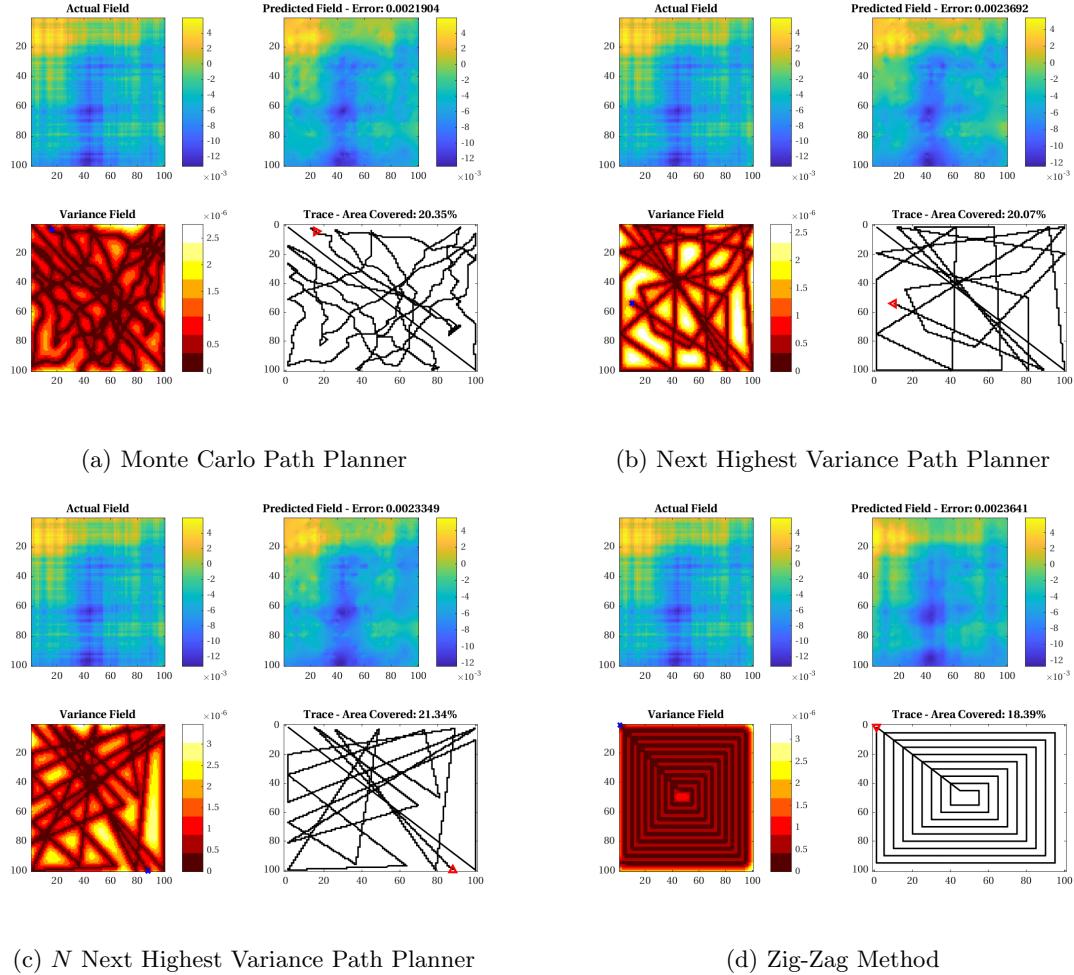
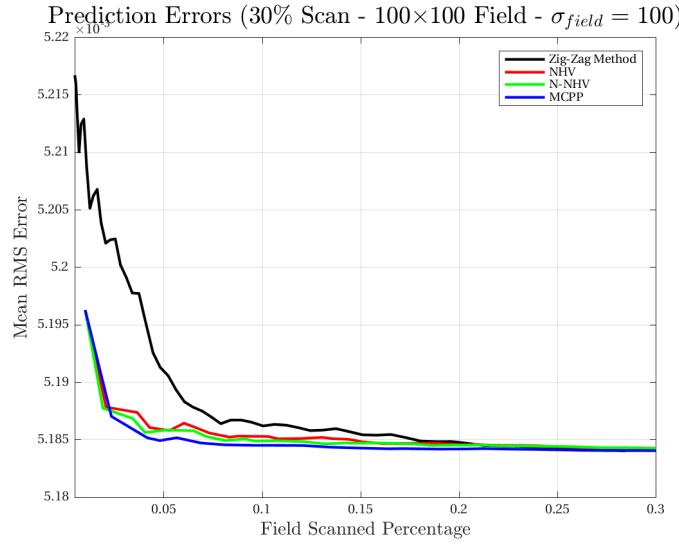


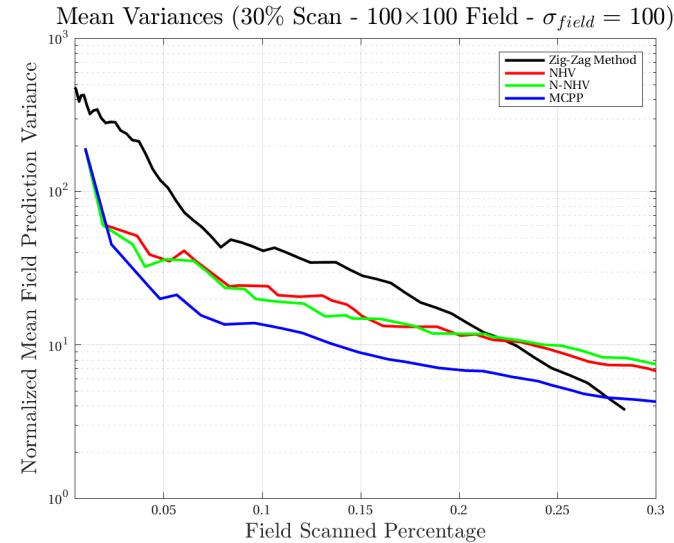
Figure A.4: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 100$, random seed: 3.

A.1.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.5: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.

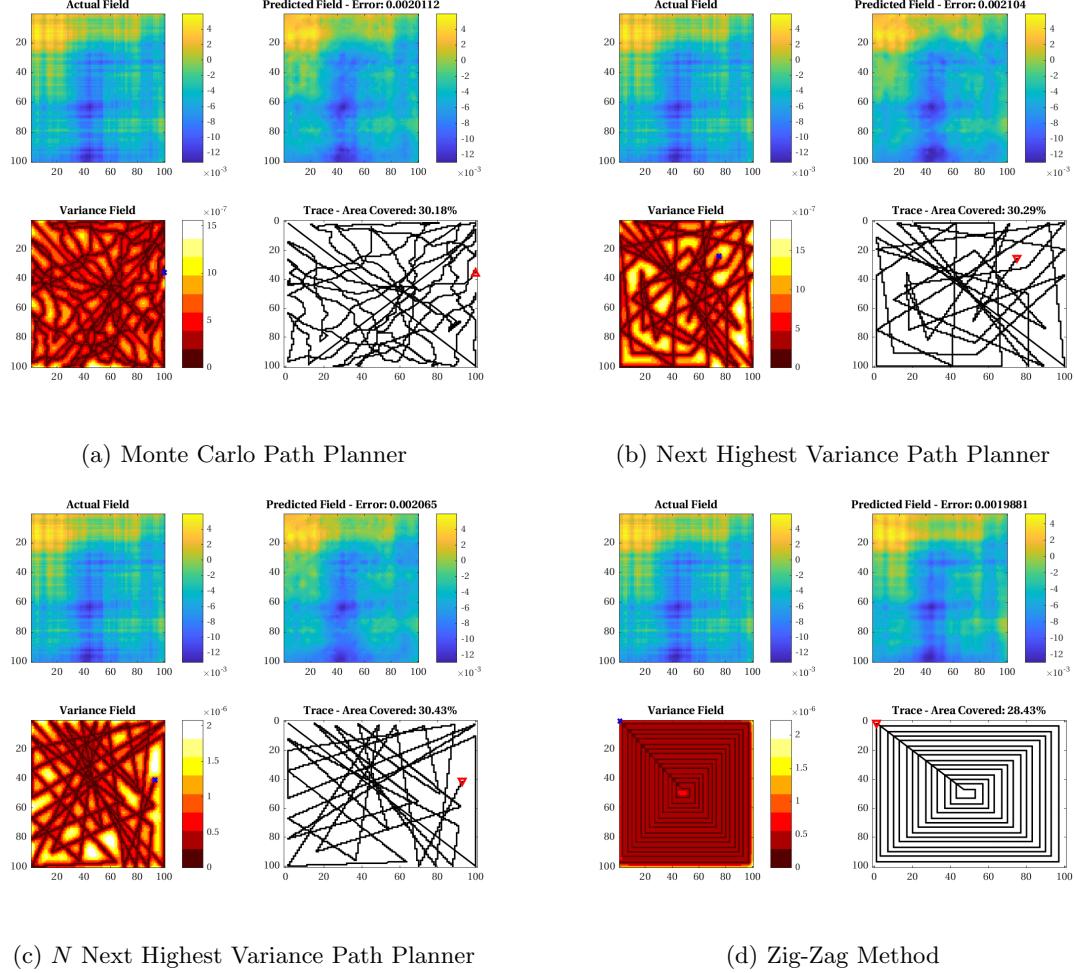
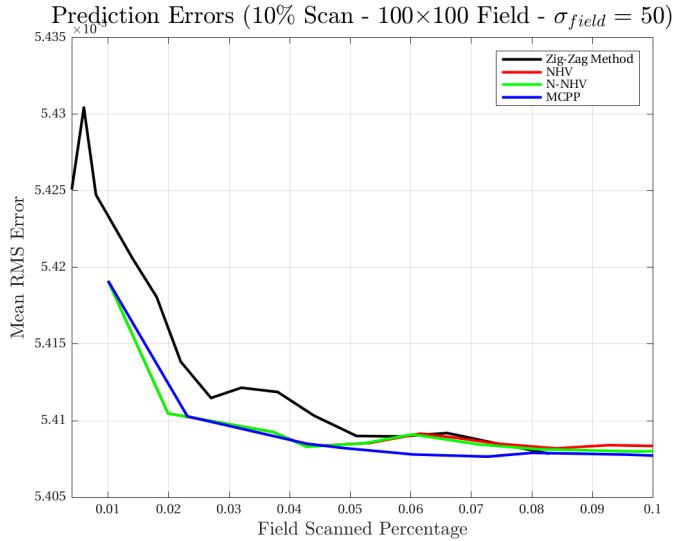


Figure A.6: Simulation output for a 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 100$, random seed: 3.

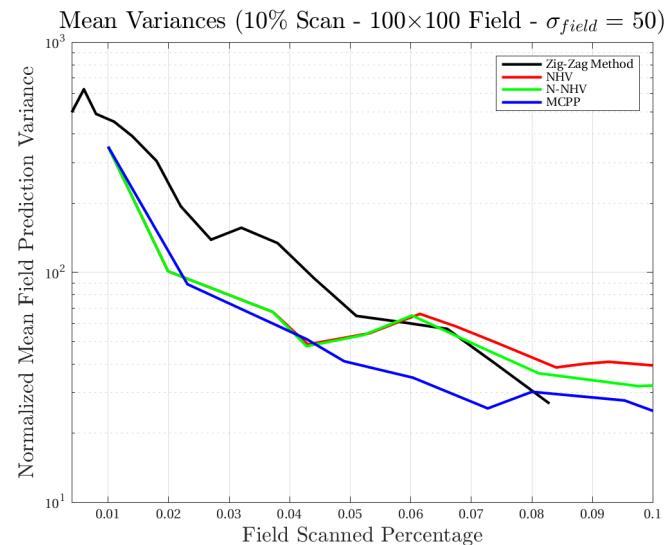
A.2 Half Width Spatial Autocorrelation Results

The methods will be compared on target fields generated with an autocorrelation factor that is half of the field width.

A.2.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.7: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.

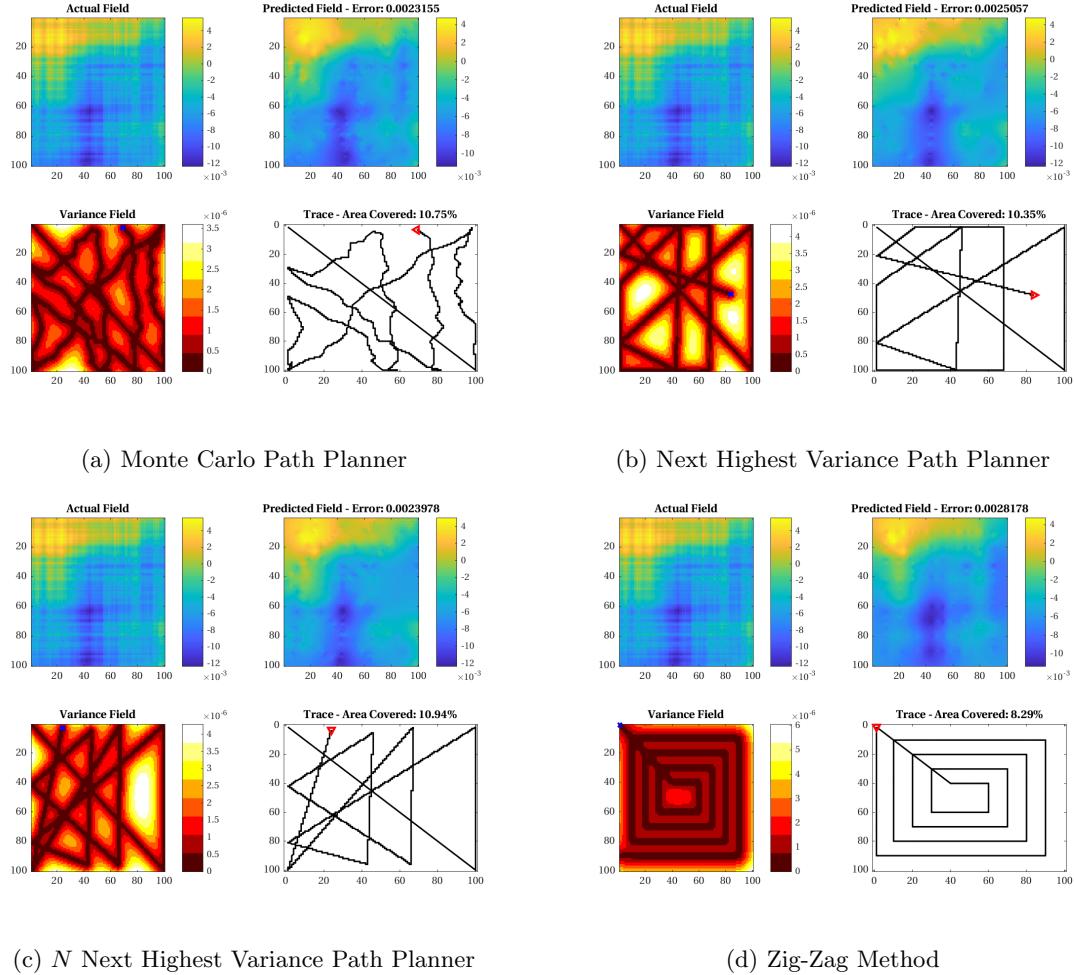
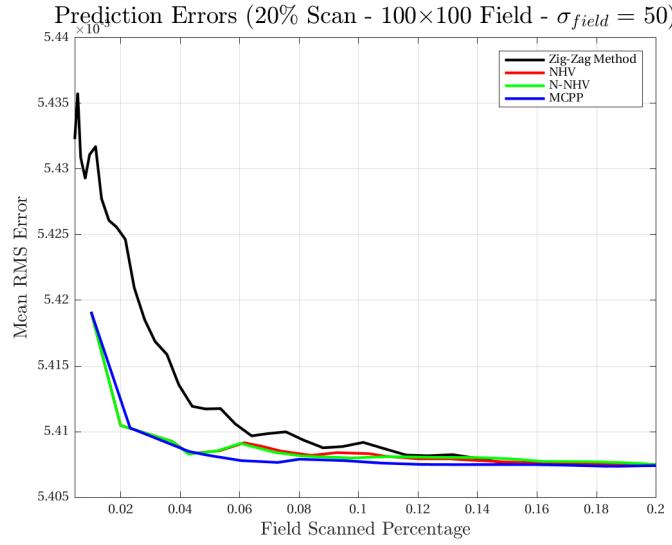


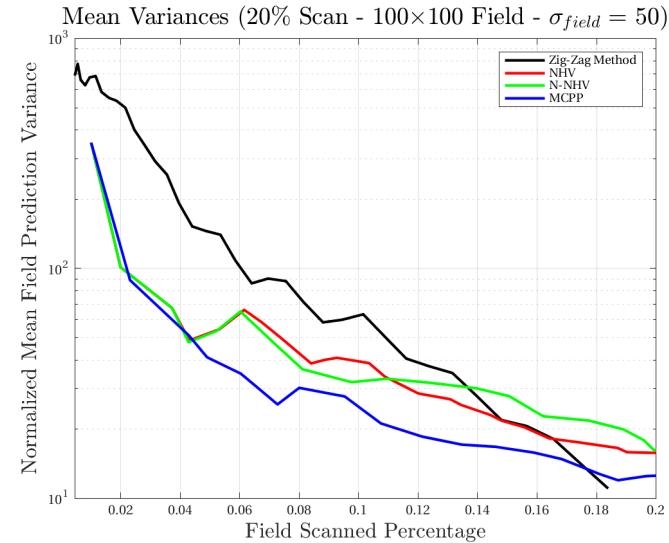
Figure A.8: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 3.

A.2.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.9: A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.

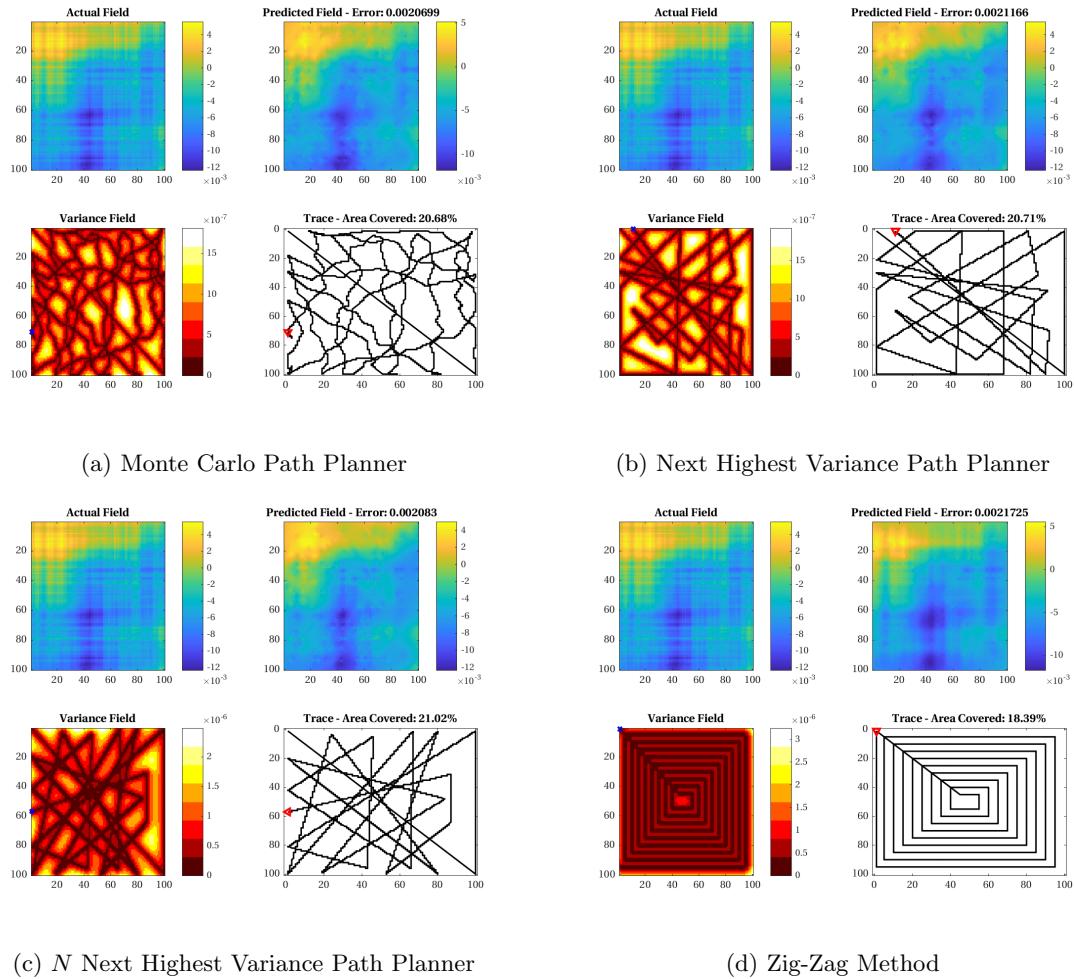
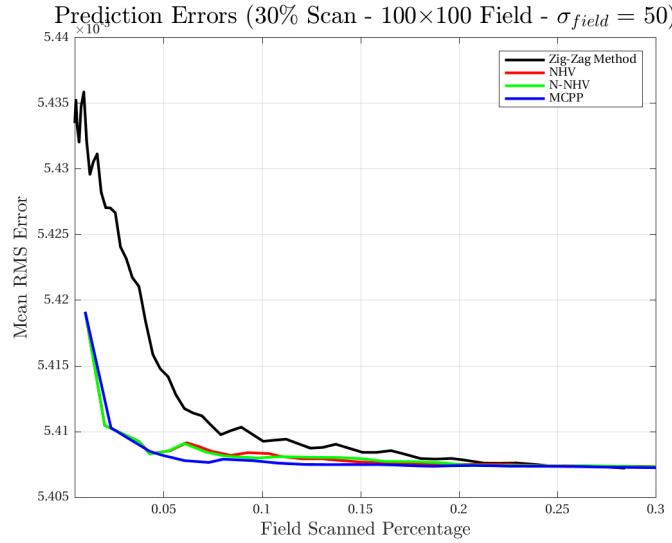


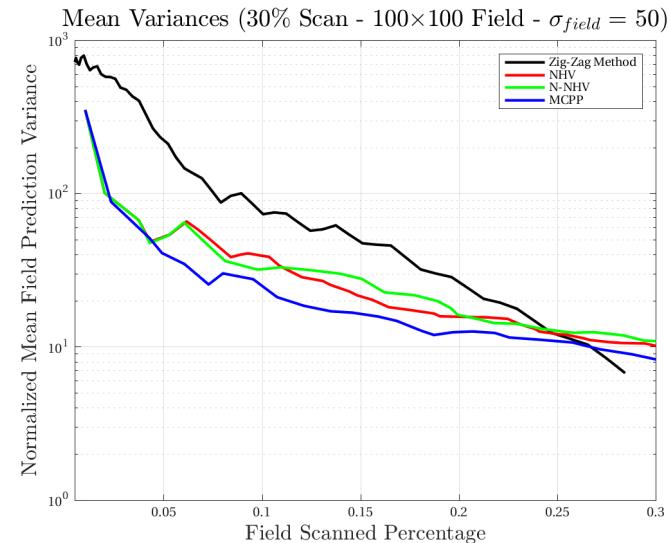
Figure A.10: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 3.

A.2.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.11: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 50$, random seed: 3.

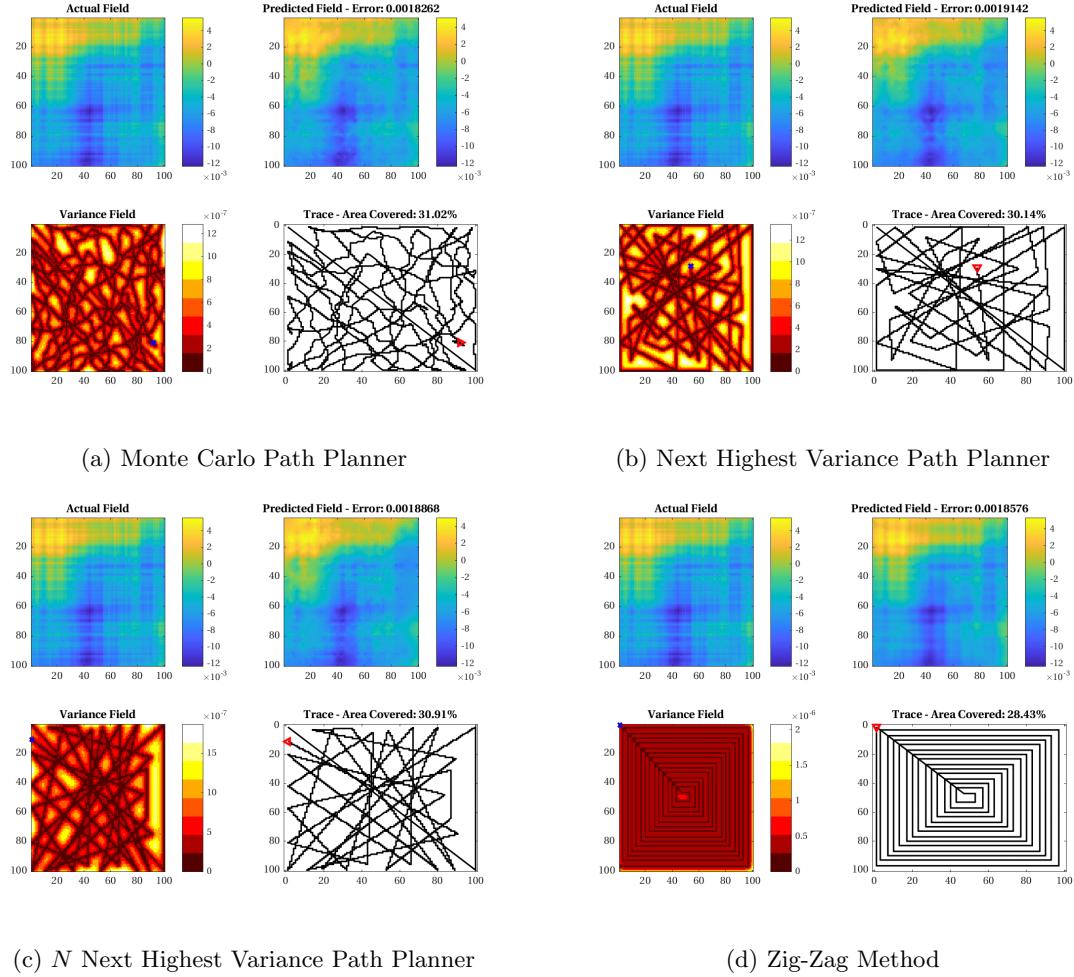


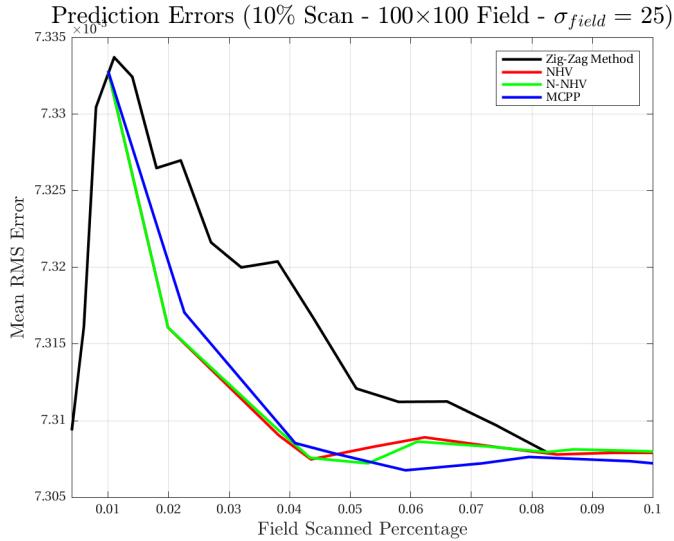
Figure A.12: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 50$, random seed: 3.

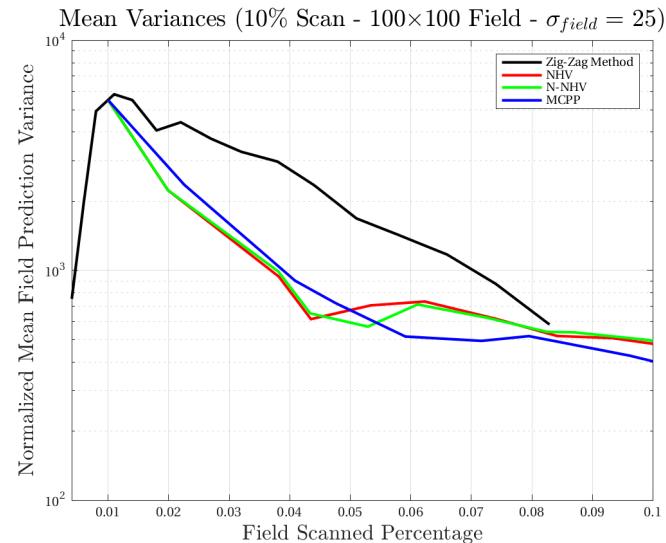
A.3 Quarter Width Spatial Autocorrelation Results

The methods will be compared on target fields generated with an autocorrelation factor that is one quarter of the field width.

A.3.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.13: A 10% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.

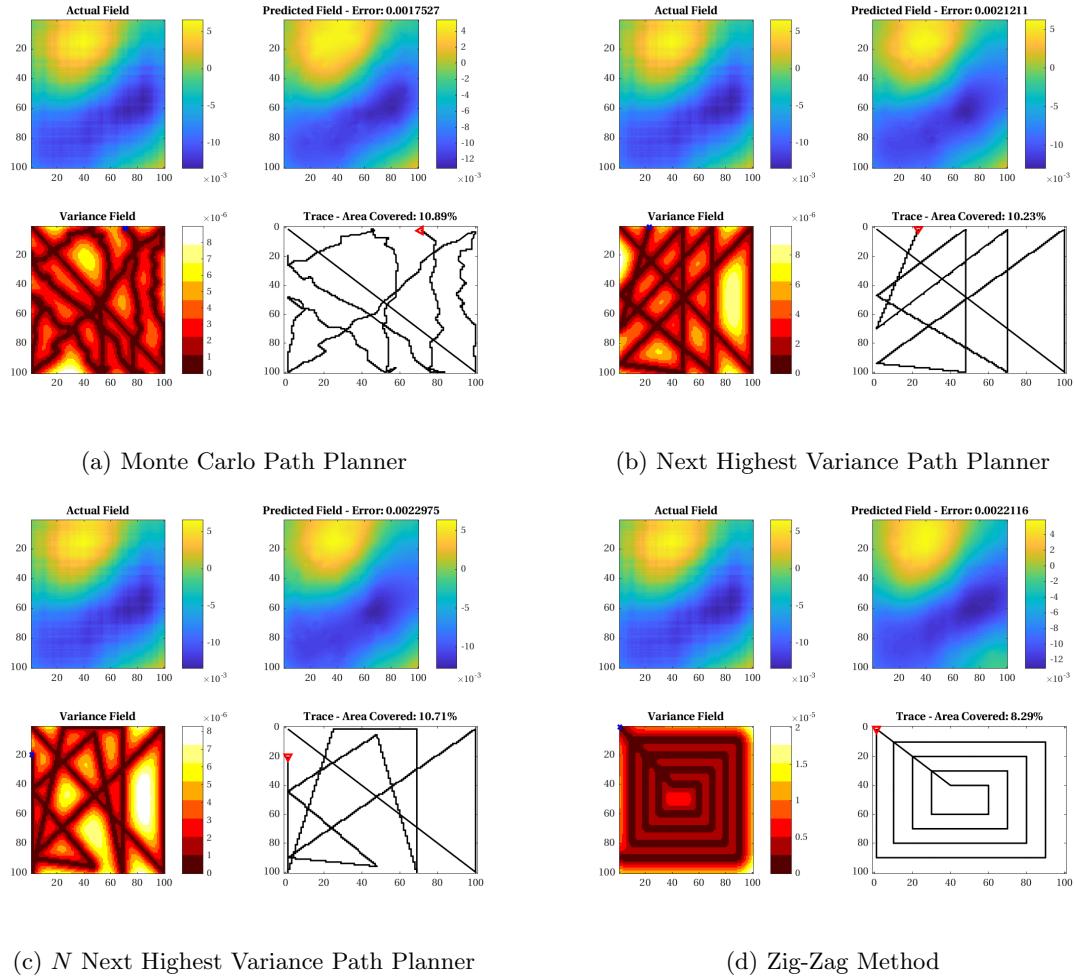
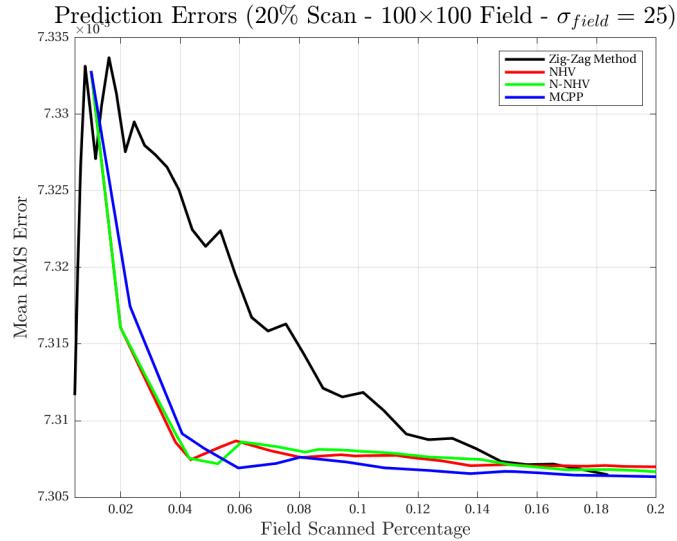


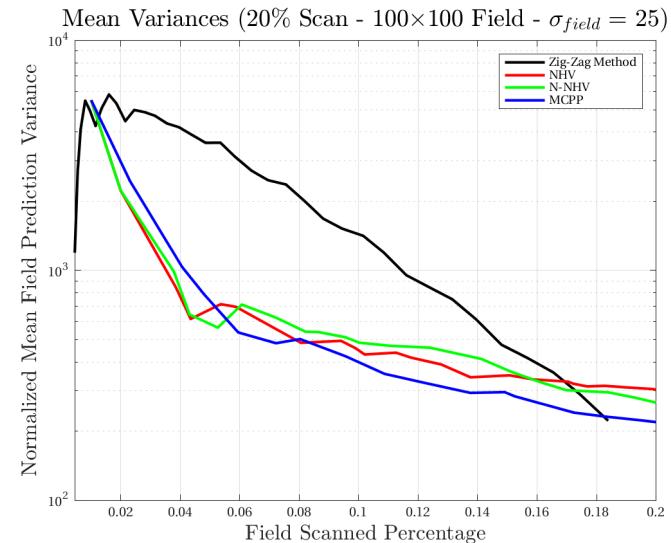
Figure A.14: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 3.

A.3.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.15: A 20% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.

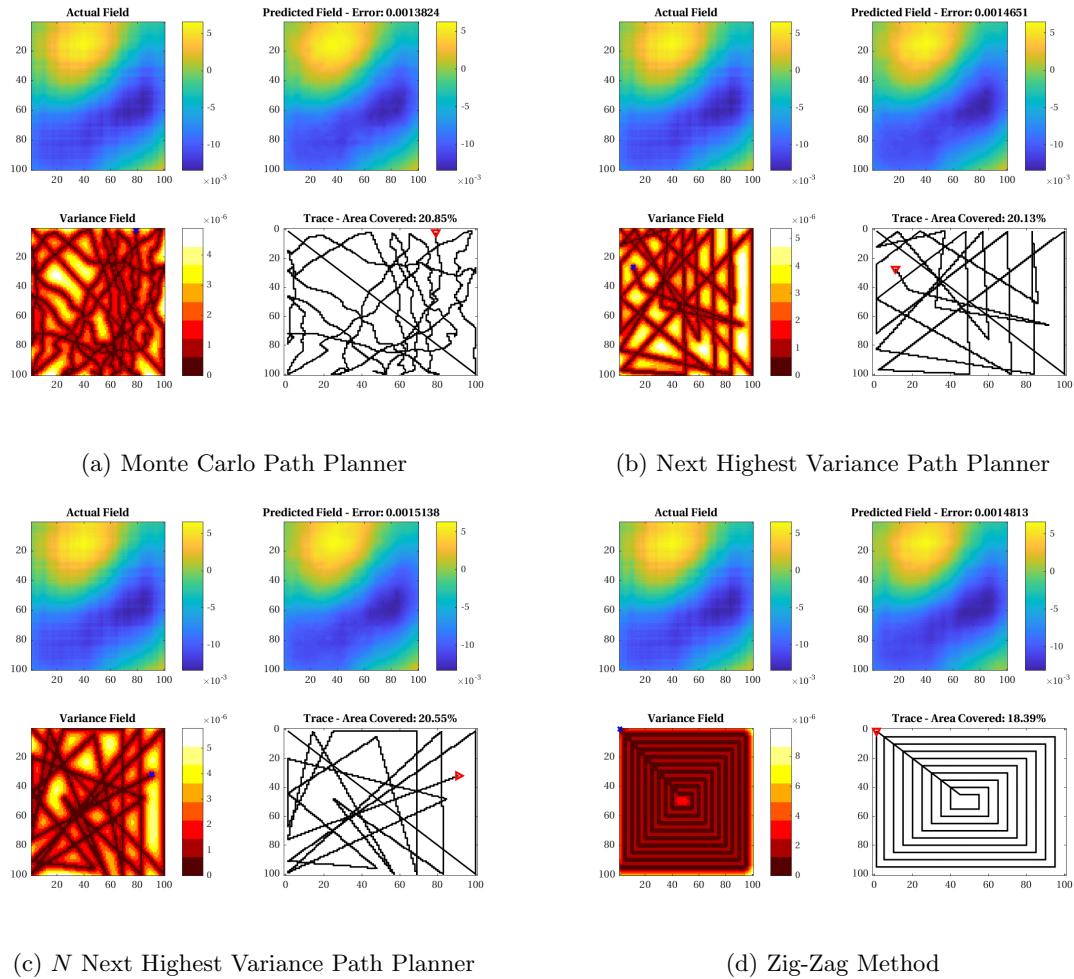
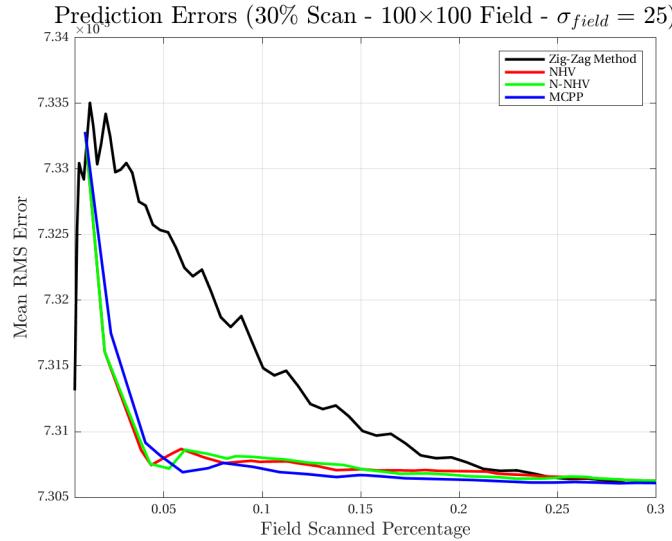


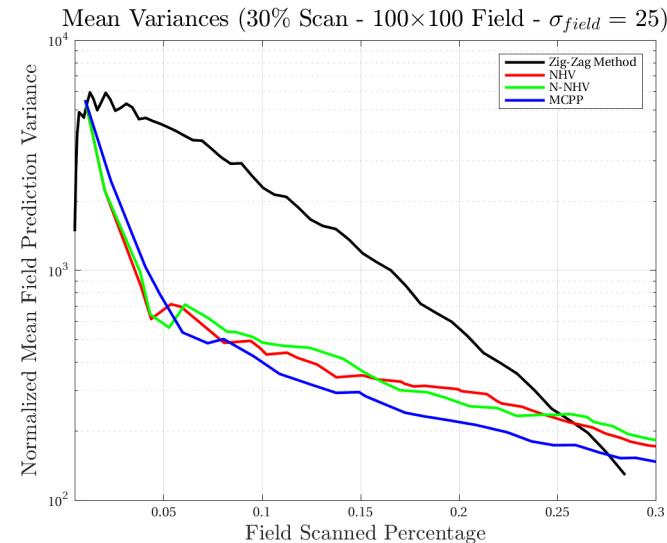
Figure A.16: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 3.

A.3.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.17: A 30% maximum area scan on a field of size 100×100 , $\sigma_{field} = 25$, random seed: 3.

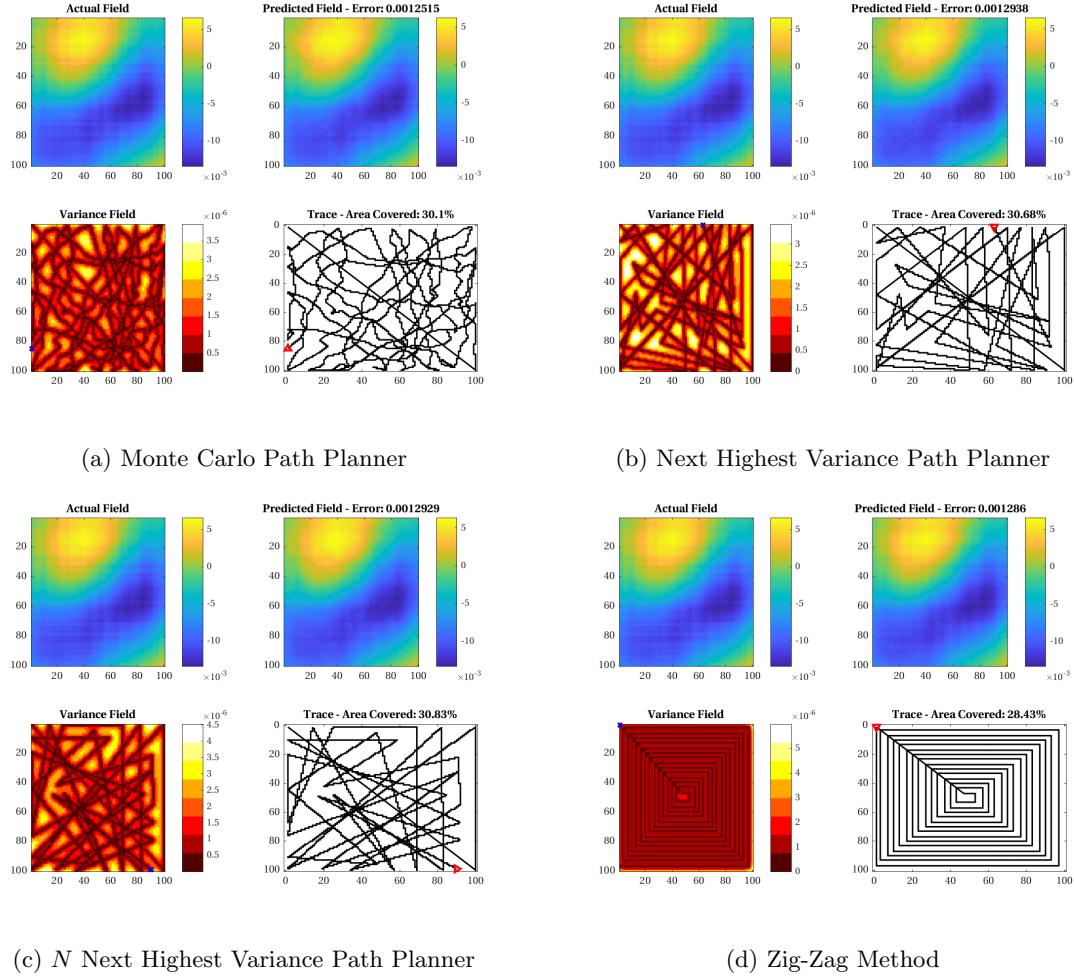


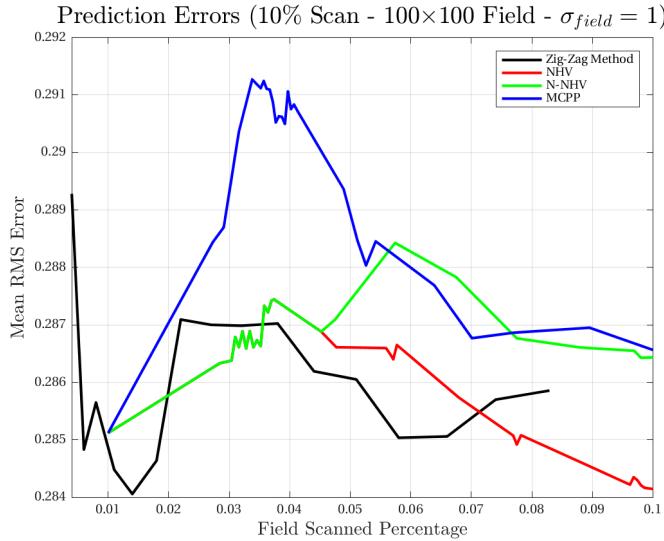
Figure A.18: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 25$, random seed: 3.

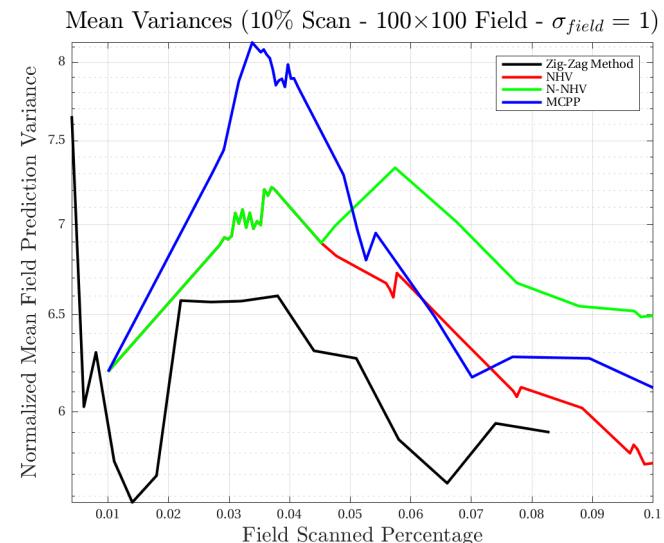
A.4 Low Spatial Autocorrelation Results

The methods will be compared on target fields generated with a low autocorrelation factor.

A.4.1 10% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.19: A 10% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 1$, random seed: 3.

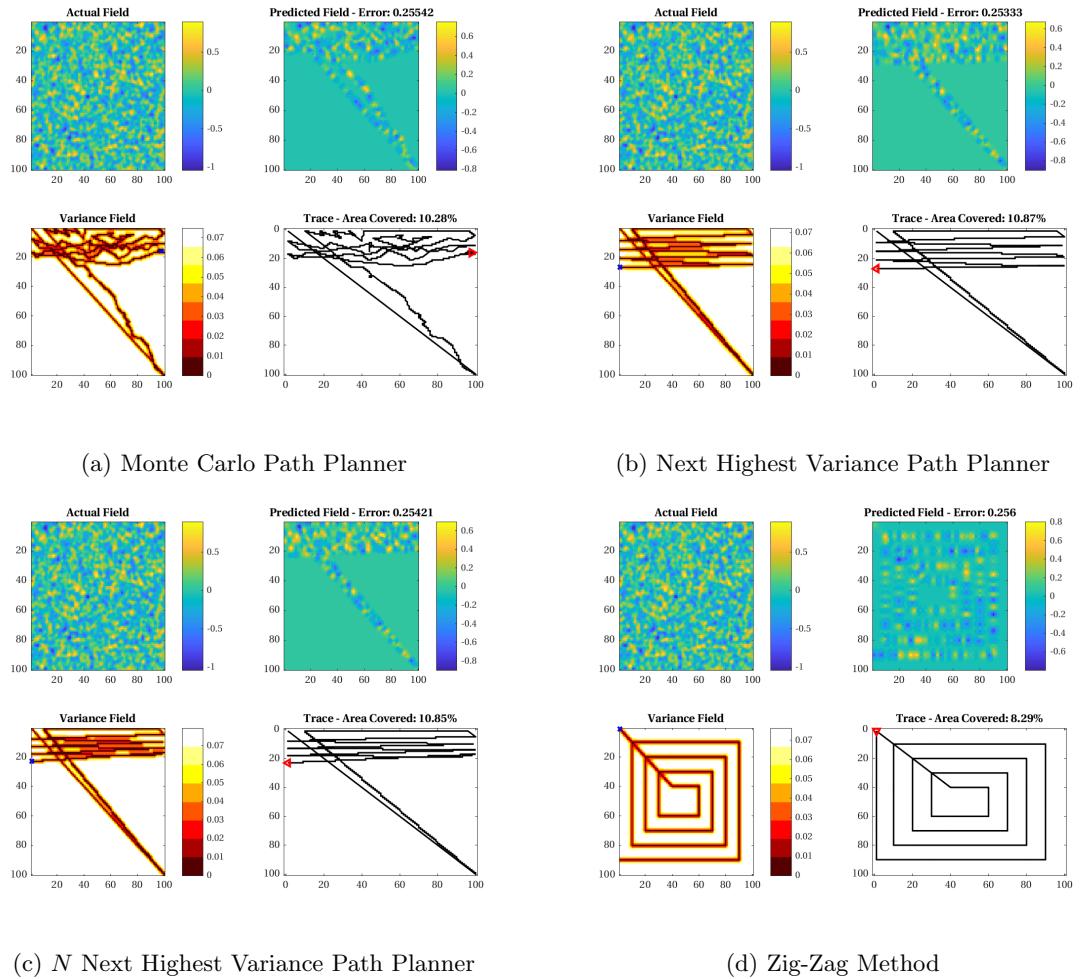
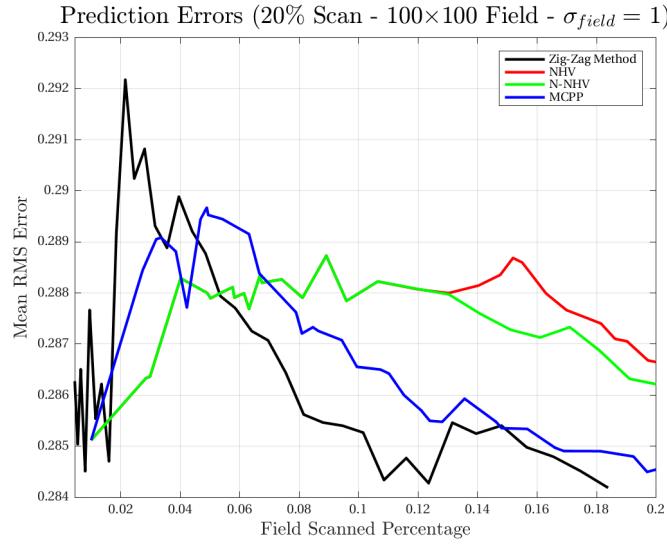


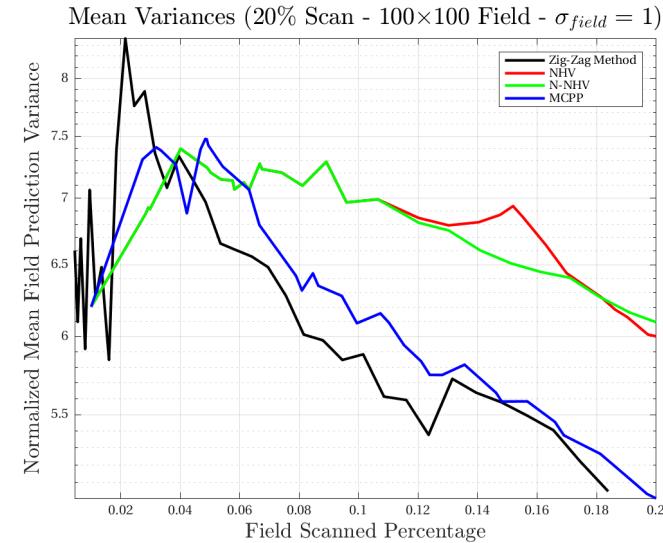
Figure A.20: Simulation output for a 10% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 3.

A.4.2 20% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.21: A 20% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 1$, random seed: 3.

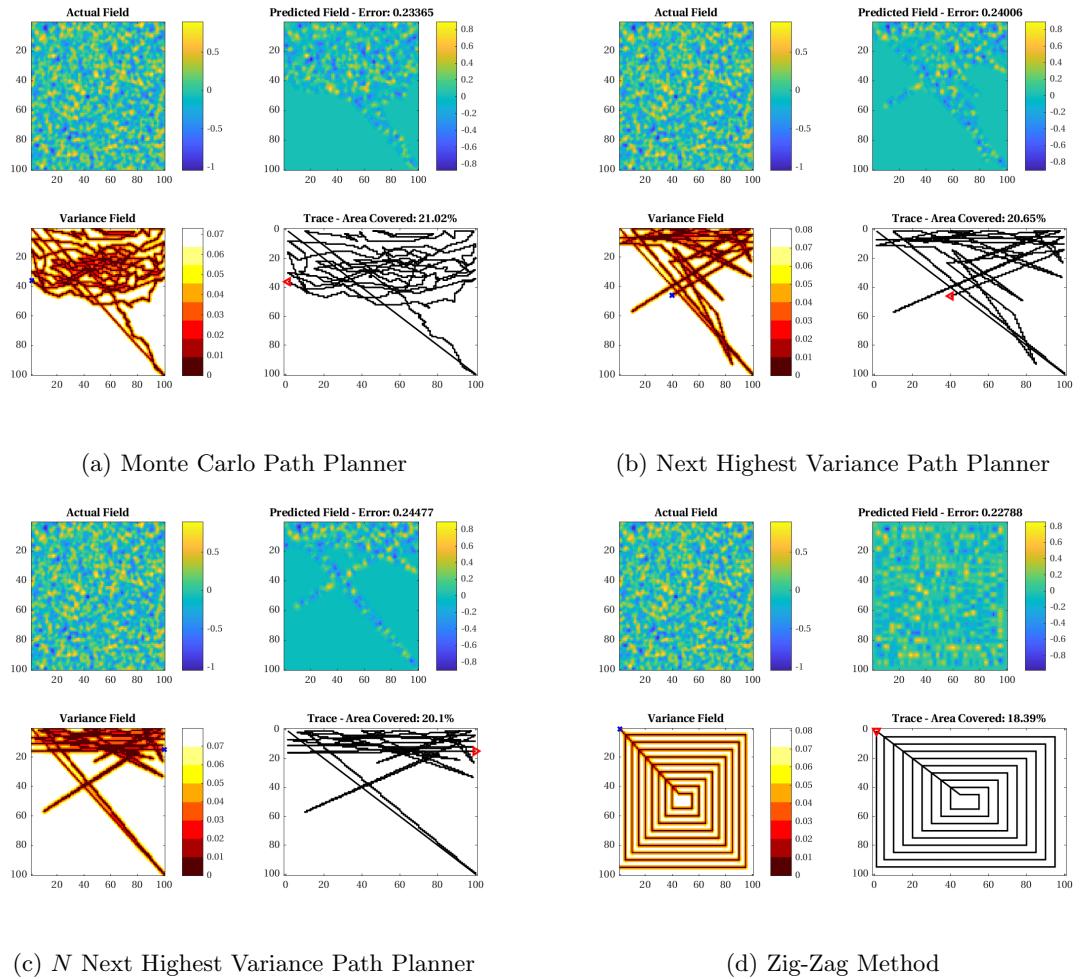
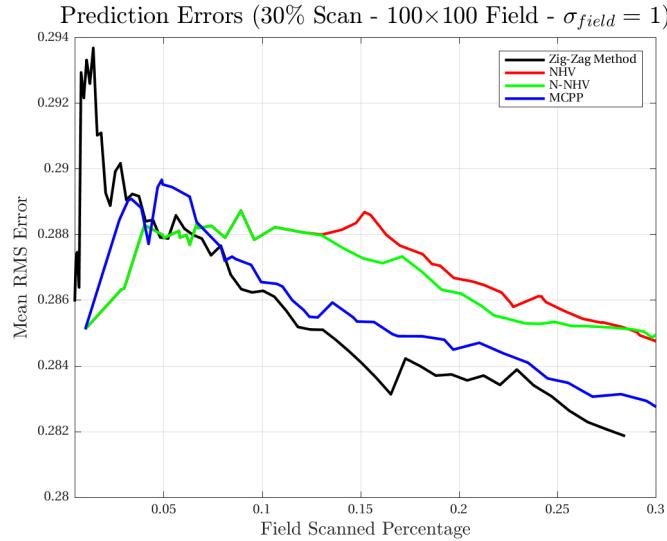


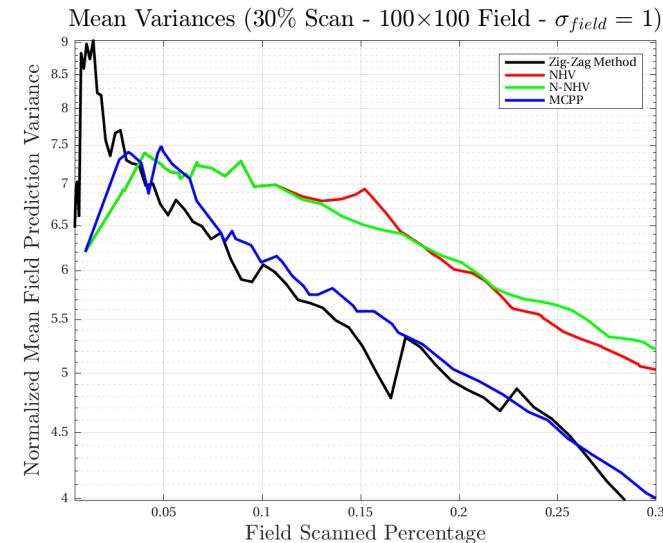
Figure A.22: Simulation output for a 20% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 3.

A.4.3 30% Maximum Field Scan



(a) Prediction errors ($\text{erf}(Z, \hat{Z})$).



(b) Semi-logarithmic prediction variances normalized to an a priori mean variance for the field.

Figure A.23: A 30% maximum area scan on a field of size 100×100 , $\sigma_{\text{field}} = 1$, random seed: 3.

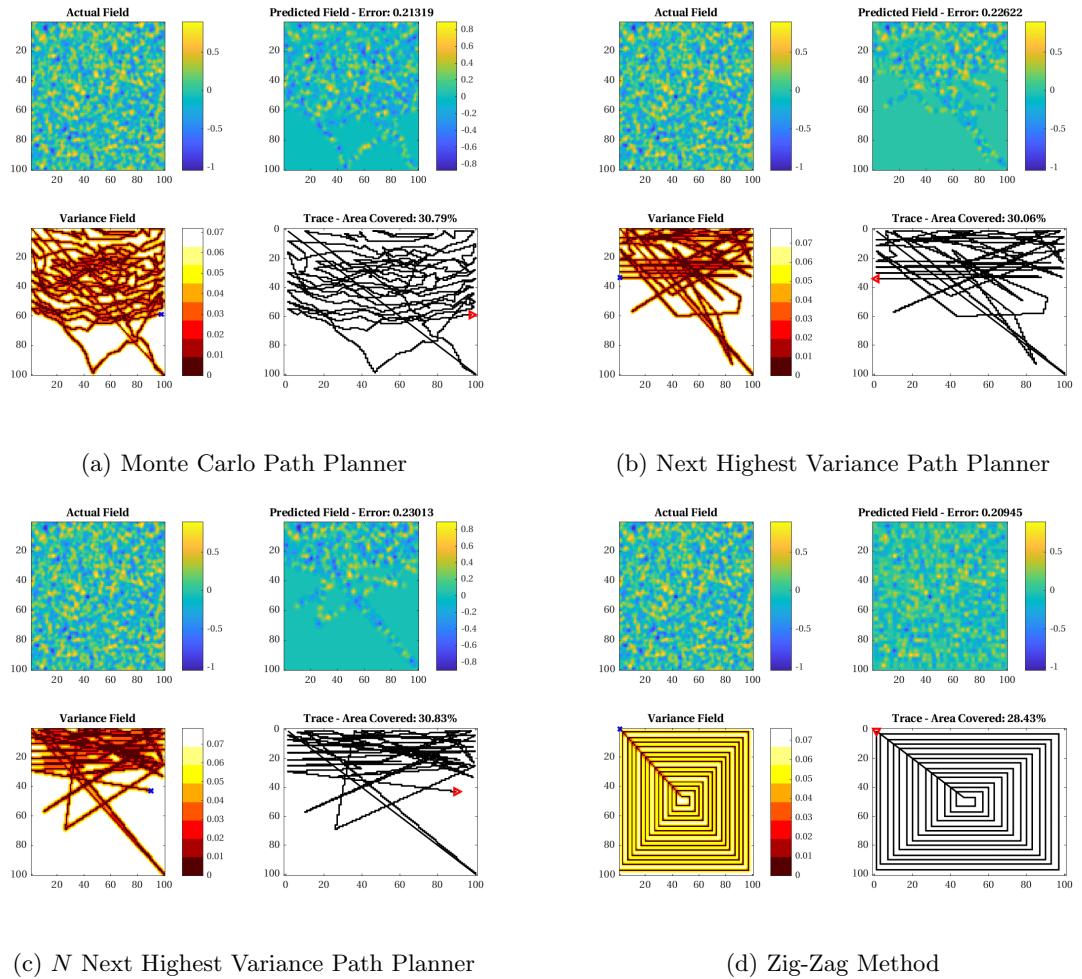


Figure A.24: Simulation output for a 30% maximum area scan on a field of size 100×100 ,

$\sigma_{field} = 1$, random seed: 3.