ROBOTIC PATH PLANNING FOR ENVIRONMENTAL FIELD ESTIMATION AND ITS APPLICATION IN AQUATIC MONITORING

by

Teng Li

B.Sc., Shandong Normal University, 2011

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

October 2018

© Teng Li, 2018

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the dissertation entitled:

Robotic Path Planning for Environmental Field Estimation and its Application in Aquatic Monitoring

submitted by	Teng Li	in partial fulfillment of the requirements for
the degree of	Doctor of Philosophy	
in	Mechanical Engineering	
Examining Co	mmittee:	
Clarence W. d	e Silva, Mechanical Enginee	ering
Supervisor		

Ryozo Nagamune, Mechanical Engineering Supervisory Committee Member

Supervisory Committee Member

Victor C.M. Leung, Electrical and Computer Engineering University Examiner

Dana Grecov, Mechanical Engineering University Examiner

Additional Supervisory Committee Members:

Farrokh Sassani, Mechanical Engineering Supervisory Committee Member

Srikantha Phani, Mechanical Engineering Supervisory Committee Member

Abstract

Recent advances in the technologies of sensing, robotics, and sensor networks have led to significant progress in environmental telemonitoring. Robotic systems have been widely developed and deployed in the field by using their capabilities of mobile sensing, autonomous navigation, and wireless communication. In particular, robotic monitoring and data sampling at locations of interest may be utilized to characterize and interpret the environmental phenomena of a study area. However, in real-world robotic sensing applications, the limitations of on-board resources will limit the coverage of the monitored area and the extent of acquired data, which will hinder the performance of field estimation and mapping. Meanwhile, the constraints of computational capability of the system components call for a computationally efficient framework to schedule and control the robotic sensing missions.

This dissertation investigates and develops systematic sensor scheduling and path planning schemes for environmental field estimation through robotic sampling, and their application in aquatic monitoring. First, a hexagonal grid-based sampling frame is designed to distribute spatially balanced sampling locations over the monitored field. Two novel hexagonal grid-based survey planners are developed to generate energy-efficient sampling paths for the exploratory survey using mobile sensing robots, which can be executed in a computationally efficient manner. Second, an energy-constrained survey planner is developed, which achieves optimal coverage density for sampling, with a limit on the energy budget. The generated survey mission guides the robots to collect data samples for estimation and mapping of an unknown field under a Gaussian Process (GP) model. Third, a hierarchical planning framework with a built-in Gaussian Markov Random Field (GMRF) model is developed to provide informative path planning and adaptive sampling for efficient spatiotemporal monitoring. Fourth, the development of a cost-effective, rapidly deployable, and easily maintainable Wireless Mobile Sensor Network (WMSN) for on-line monitoring of surface water is presented. A novel On-Line Water Quality Indexing (OLWQI) scheme is developed and implemented to interpret the large volume of on-line measurements.

The experimental results in the present dissertation demonstrate the effectiveness and efficiency of the proposed planning schemes and their application in aquatic environmental monitoring.

Lay Summary

Robotic systems provide efficiency and flexibility for the sensing process in remote environmental monitoring. This dissertation addresses the scheduling and planning issues when applying robotic sampling in environmental monitoring, particularly in aquatic monitoring.

First, the problem of regular grid-based survey planning is studied, which will navigate the robots to measure the sampling locations for an environmental survey. Second, the survey planners are improved to tackle a real-world problem where the system operation is limited to the available power supply and energy budget. Third, an informative path planner is developed to improve the feasibility and efficiency of the mobile sensing process, which determines the more informative locations. Furthermore, a wireless mobile sensor network is designed and developed for on-line monitoring of surface water. A novel on-line water quality index is introduced to interpret the complex measurements as a simple quality indicator.

Preface

All the results presented in this dissertation was conducted based on my original ideas and work in the Industrial Automation Laboratory (IAL) at the University of British Columbia (UBC), Vancouver campus, under the direct supervision and guidance of Dr. Clarence W. de Silva, Professor in the Department of Mechanical Engineering at UBC. Dr. de Silva proposed and supervised the overall research project, suggested the research topics, obtained funding, provided research facilities and field test opportunities, and revised the dissertation and other publications. Parts of the work in the dissertation have been or will be published in peer-reviewed journals and conferences, as listed below.

The results in Chapter 3, 4 and 6 have been published:

- Teng Li, Min Xia, Jiahong Chen, Yuanjie Zhao, and Clarence W. de Silva, "Automated Water Quality Survey and Evaluation Using an IoT Platform with Mobile Sensor Nodes," *Sensors*, vol. 17, no. 8, p. 1735, 2017. I conceived all the algorithms and developed the platform. I also analyzed the data and wrote the manuscript. Dr. Min Xia and Mr. Jiahong Chen helped in performing the field tests. Mr. Yuanjie Zhao developed the graphic user interface. Dr. Clarence W. de Silva suggested the research topic and procedures, acquired and provided funding and laboratory facilities for the work, continuously reviewed and advised on the research activities, suggested research directions and possible approaches, guided the preparation of the manuscript, and carried out revisions and editing of the paper.
- Teng Li, Min Xia, Jiahong Chen, Shujun Gao, and Clarence W. de Silva, "A Hexagonal Grid-based Sampling Planner for Aquatic Environmental Monitoring using Unmanned Surface Vehicles," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 3683-3688, 2017. I conducted all the algorithms and wrote the manuscript. Dr. Min Xia and Mr. Jiahong Chen helped in performing the field tests. Dr. Shujun Gao offered some advice for the data analysis. Dr. Clarence W. de Silva suggested the research topic and procedures, acquired and provided funding and laboratory facilities for the work, continuously reviewed and advised on the research activities, suggested

research directions and possible approaches, directed the work, guided the preparation of the manuscript, and carried out revisions and editing of the paper. The results in Chapter 3 and 4 have been submitted to a peer reviewed journal:

• Teng Li, Chaoqun Wang, Bing Li, Max Q.-H. Meng, and Clarence W. de Silva, "Mobile Sensor Scheduling for Environmental Exploration and Field Mapping," 2018 (under review). I developed the ideas and algorithms of this work, and wrote the manuscript. Mr. Chaoqun Wang helped in conducting the simulations. Dr. Bing Li and Dr. Max Q.-H. Meng provided important suggestions for improving the paper. Dr. Clarence W. de Silva suggested the research topic and procedures, acquired and provided funding and laboratory facilities for the work, continuously reviewed and advised on the research activities, suggested research directions and possible approaches, directed the work, guided the preparation of the manuscript, and carried out revisions and editing of the paper.

The results in Chapter 5 have been published or to be submitted for publication:

- Teng Li, Kaitai Tong, Min Xia, and Clarence W. de Silva, "Adaptive Sampling for Environmental Field Estimation with Robotic Sensors," 2018 (to be submitted). The main algorithms, theoretical verifications, and experiments are based on my original ideas. Kaitai Tong contributed to the simulation. Dr. Min Xia provided important suggestions. Dr. Clarence W. de Silva suggested the research topic and procedures, acquired and provided funding and laboratory facilities for the work, continuously reviewed and advised on the research activities, suggested research directions and possible approaches, guided the preparation of the manuscript.
- Chaoqun Wang, Teng Li, Max Q.-H. Meng, and Clarence W. de Silva, "Efficient Mobile Robot Exploration with Gaussian Markov Random Fields in 3D Environments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5015-5021, 2018. Mr. Chaoqun Wang contributed the original idea and wrote the manuscript. I provided important advice to the preparation of the paper and helped in revising the manuscript. Dr. Max Q.-H. Meng and Dr. Clarence W. de Silva suggested the research topic and procedures, provided funding and laboratory facilities for the work, continuously reviewed on the research activities, guided the preparation and revision of the manuscript.

Table of Contents

Abstract	iii
Lay Summ	naryv
Preface	vi
Table of C	Contents viii
List of Ta	bles xi
List of Fig	gures xii
List of Syr	mbols xiv
List of Ab	breviations xvii
Acknowle	dgements xix
Chapter 1	: Introduction1
1.1 N	Motivation1
1.2 H	Background and Challenges
1.3 I	Research Objectives and Contributions
1.4 (Organization of the Dissertation
Chapter 2	: Related Work
2.1 A	Automated Environmental Monitoring
2.2 H	Field Sampling
2.2.1	Sampling Design
2.2.2	Exploratory Survey 10
2.2.3	Coverage Path Planning for Sampling11
2.3 H	Environmental Field Estimation
2.3.1	Random Fields
2.3.2	Gaussian Processes
2.3.3	Gaussian Process Regression
2.3.4	Gaussian Markov Random Fields
2.4 A	Adaptive Sampling
2.5 0	Quality Indexing 17
Chapter 3	: Grid-based Survey Planning18
3.1 0	Dverview
	viii

3.2	Coverage Sampling Design	19
3.3	Hexagonal Grid-based Survey Planning	20
3.3	.1 Spanning Tree-based Coverage Survey Planner	21
3.3	2 Hexagonal Grid-based Coverage Survey Planner	
3.3	.3 Sub-paths for Multiple Mobile Sensors	34
3.4	Experimental Results	36
3.5	Summary	42
Chapter	• 4: Energy-constrained Exploratory Sampling	43
4.1	Overview	43
4.2	Random Field and Environmental Model	45
4.3	Energy-Constrained Exploratory Sampling	46
4.3	.1 Optimal Density of Coverage Sampling	48
4.3	2 Model Estimation and Field Mapping	49
4.4	Experiments and Discussion	51
4.4	1 Dataset Setup	52
4.4	2 Estimation Performance	53
4.4	.3 Computational Cost	59
4.5	Field Test	62
4.6	Summary	64
Chapter	5: Informative Path Planning	65
5.1	Overview	65
5.2	Problem Statement	67
5.3	Hierarchical Informative Path Planner	68
5.3	1 Local Greedy Planner	69
5.3	2 Global Planner of Near-Optimal Design	71
5.4	Numerical Experiments and Discussion	80
5.5	Summary	88
Chapter	: 6: On-line Monitoring of Surface Water	89
6.1	Overview	89
6.2	On-line Water Quality Index	90
6.3	Development of a WMSN for On-line Monitoring of Surface Water	97
		ix

6.3.1	Mobile Sensor Nodes	
6.3.2	2 Base Station and Remote Server	100
6.4	Case Study	101
6.4.1	Real-World Dataset	102
6.4.2	2 Field Test	
6.5	Summary	108
Chapter	7: Conclusions and Future Work	109
7.1	Conclusions	109
7.2	Possible Future Work	111
Referenc	es	113
Appendi	X	122
Appen	dix A	122
A.1	Proof of Proposition 5.3	122
Appen	dix B	124
B .1	Proof of the Factor Sensitivity	124

List of Tables

Table 3.1: MST Vertex Creation.	22
Table 3.2: Algorithm performance of survey planning.	37
Table 4.1: Algorithm Performance of Field Estimation and Mapping.	55
Table 4.2: RMSE performance of different survey planners in the field tests	63
Table 5.1: Target sites and their corresponding assignments.	87
Table 6.1: Factor sensitivity of the CCME WQI.	94
Table 6.2: Factor sensitivity of the Online Water Quality Index (OLWQI)	96
Table 6.3: Indexing results using data collected at the First Landing Station in C	BIBS.
	102
Table 6.4: Stepwise regression analysis: final index versus three factors.	103
Table 6.5: Statistical results of the factor effects on the final index	104

List of Figures

Figure 3.1: Hexagonal tessellation and design of coverage sampling with SLoIs	20
Figure 3.2: Coarse cell tessellation and relative constructed MST	23
Figure 3.3: Possible MST vertices and edges surrounding a coarse cell	24
Figure 3.4: Generation of the circumnavigation path based on the MST	26
Figure 3.5: Strategies of path segment generation	27
Figure 3.6: Overall path cycle after path update	27
Figure 3.7: Final planned path for coverage sampling	27
Figure 3.8: Edge candidates of neighboring SLoI pairs	29
Figure 3.9: Rules of path segment generation in an ACC.	30
Figure 3.10: Generated path segments to visit full ACCs.	31
Figure 3.11: Combination of neighboring cycles.	31
Figure 3.12: Generated coverage path after incorporating all unvisited SLoIs	31
Figure 3.13: Final planned path for coverage sampling	32
Figure 3.14: Aerial view of the Yosef Wosk Reflecting Pool.	36
Figure 3.15: Field deployment of a USV	36
Figure 3.16: Generated survey plans with respect to different sampling resolutions	39
Figure 3.17: Algorithm performance of processing time	40
Figure 3.18: Evaluation results of the survey mission	41
Figure 4.1: Edge candidates of neighboring SLoI pairs for path planning	46
Figure 4.2: Ground truth map of the study area	52
Figure 4.3: Algorithm performance in field estimation and mapping for different per	ower
supply settings	54
Figure 4.4: Results of sampling locations, sampling paths, and mapping results u	sing
different survey planners	57
Figure 4.5: Results of sampling locations, sampling paths, and results of estimated	ation
uncertainty using different survey planners	58
Figure 4.6: Relationships between cell size and number of SLoIs, and total length	59
Figure 4.7: Algorithm performance related to computational efficiency.	61
Figure 4.8: Results of the proposed HGC survey planner.	63
	xii

Figure 5.1: Generation of the next sampling location on the local path	70
Figure 5.2: Execution example of local path planning.	71
Figure 5.3: Next sampling location scheduled by the local planner	73
Figure 5.4: Bounded area of possible sampling locations	74
Figure 5.5: Two consecutive sampling locations	75
Figure 5.6: Dumbbell-shaped area	77
Figure 5.7: Flow chart of the hierarchical informative path planner.	79
Figure 5.8: Numerically generated spatial field	80
Figure 5.9: Triangle mesh of the discretized spatial field	81
Figure 5.10: RMSE performance of resulting predictions	82
Figure 5.11: Mapping results of the proposed planner at different time (second)	83
Figure 5.12: Mapping results of the DPP-LGS approach at different time (second)	84
Figure 5.13: Algorithm performance of the CSP-GO approach	85
Figure 5.14: Generated target sites and sampling paths at time $t = 900s$	86
Figure 6.1: The proposed wireless mobile sensor network	97
Figure 6.2: The design framework of the MSN	99
Figure 6.3: The developed MSN in the wireless mobile sensor network	100
Figure 6.4: The design architecture of the BS.	101
Figure 6.5: The developed BS in the sensing platform	101
Figure 6.6: Field deployment of the platform at the Yosef Wosk Reflecting Pool	104
Figure 6.7: Graphical User Interface (GUI)	105
Figure 6.8: Comparison of the CCME WQI and the OLWQI	107

List of Symbols

A	Study area
Â	Contour of study area
Α	Projector matrix
В	Boolean function
b	Boolean variable
$\mathcal C$	Number of hexagonal grid centroids
d	Distance between neighboring sampling locations
$d_{\mathbf{s}_i,\mathbf{s}_j}$	Distance between two locations
E_T	Set of edges of minimum spanning tree
$e_{_{bdt}}$	Energy budget
$e(d_{\mathbf{s}_i,\mathbf{s}_j})$	Energy cost when traveling between two locations
$e_{_M}$	Energy cost of measurement process
F_{i}	Quality index factor
P	Number of samples involved in index calculation
f	
^f G	Guideline matrix
\mathbf{G} $\mathcal{GMRF}(\cdot)$	Guideline matrix Distribution of a Gaussian Markov Random Field
\mathcal{G} $\mathcal{GMRF}(\cdot)$ $\mathcal{GP}(\cdot)$	Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process
$G \\ \mathcal{GMRF}(\cdot) \\ \mathcal{GP}(\cdot) \\ I$	Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation
G $\mathcal{GMRF}(\cdot)$ $\mathcal{GP}(\cdot)$ I $ I $	Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes
G $\mathcal{GMRF}(\cdot)$ $\mathcal{GP}(\cdot)$ I $ I $ l	Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell
G $\mathcal{GMRF}(\cdot)$ $\mathcal{GP}(\cdot)$ I $ I $ l ℓ	 Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell Number of monitored parameters
F G GMRF(·) GP(·) I I l ℓ M	 Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell Number of monitored parameters Number of mobile sensing robots
F G GMRF(·) GP(·) I I l ℓ M N	 Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell Number of monitored parameters Number of mobile sensing robots Normal distribution
F G GMRF(·) GP(·) I I l ℓ M N N N	 Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell Number of monitored parameters Number of mobile sensing robots Normal distribution Number of sampled locations
F G GMRF(·) GP(·) I I l ℓ M N N N P	 Guideline matrix Distribution of a Gaussian Markov Random Field Distribution of a Gaussian process Set of grid nodes for spatial interpolation Number of the interpolation grid nodes Edge length of a hexagonal cell Number of monitored parameters Number of mobile sensing robots Normal distribution Number of sampled locations Number of failed parameters

p	Total path length
Q	Precision matrix
\mathcal{Q}	Number of failed tests
$\mathbb R$	Set of real numbers
S	Set of sampled locations
\widetilde{S}	Set of possible locations within study area
SL	Set of sampling locations of interest
SL	Number of sampling locations of interest
SN	Set of unvisited sampling locations
SN	Number of unvisited sampling locations
SP	Set of sampling locations on path
SP	Number of sampling locations on path
S	Sampling location within study area
S	Factor sensitivity
Т	Random process of environmental field
$T_{ m min}$	Minimum spanning tree
t	Time variable
t_M	Time cost of measuring process
t_p	Time cost of completing a path
$t_{\mathbf{s}_i,\mathbf{s}_j}$	Traveling time cost between two locations
t_{udt}	Time interval for measurement update
\mathcal{T}	Current time
U	Set of mobile sensor nodes
и	Number of mobile sensor nodes
V_{T}	Set of vertices of minimum spanning tree
v	Average speed of a mobile sensor node
v_T	Vertex of minimum spanning tree
X	Regression function

Y	Gaussian process
Y	Random vector of measurements
У	Realization matrix of measurements
\mathbb{Z}^+	Set of positive integers
μ	Mean function
Σ	Covariance function
σ^2	Marginal variance
σ_{ε}^2	Noise variance

List of Abbreviations

ACC	Auxiliary Coarse Cell
AUV	Autonomous Underwater Vehicles
BS	Base Station
CAU	Central Assessment Unit
CCME	Canadian Council of Ministers of Environment
DMPP	Discrete Monotone Polygonal Partitioning
GMRF	Gaussian Markov Random Field
GP	Gaussian Process
GUI	Graphic User Interface
LAU	Local Assessment Unit
LiPo	Lithium Polymer
MSN	Mobile Sensor Node
MST	Minimum Spanning Tree
NOAA	National Oceanic and Atmospheric Administration
NP	Non-deterministic Polynomial
OLWQI	Online Water Quality Index
OSDR	Optimal Sampling Density Regulator
PVC	Polyvinyl chloride
RS	Remote Server
SD	Sampling Design
SAR	Sub-Area Removal
SDR	Sampling Density Regulator
SLoI	Sampling Location of Interest
SLoR	Sampling Location of Region
SGSTC	Square Grid-based Spanning Tree Coverage
SPDE	Stochastic Partial Differential Equation
TSP	Travelling Salesman Problem
UAV	Unmanned Aerial Vehicle
USV	Unmanned Surface Vehicle

WMSN Wireless Mobile Sensor Network

WSN Wireless Sensor Network

WQI Water Quality Index

Acknowledgements

First I wish to thank my supervisor Prof. Clarence W. de Silva. His rigorous supervision, valuable suggestions, and strong support enabled me to successfully and rewardingly complete this dissertation and my Ph.D. study at the University of British Columbia. I am grateful to him for offering me the opportunity to pursue my Ph.D. degree under his direct supervision. I deeply appreciate and admire him for his expertise, guidance, and inspiration that will undoubtedly benefit me throughout my personal development and professional career.

Acknowledgement should be made to all of my colleagues at the Industrial Automation Laboratory over the years: Dr. Haoxiang Lang, Dr. Yanjun Wang, Dr. Yunfei Zhang, Dr. Muhammad Tufail, Dr. Min Xia, Dr. Shujun Gao, Dr. Yu Du, Dr. Lili Meng, Mr. Hani Balkhair, Mr. Shan Xiao, Mr. Zhuo Chen, Mr. Fan Yang, Mr. Tongxin Shu, Mr. Sheikh Tanvir, Mr. Jiahong Chen, Mr. Bilal Riaz, Mr. Lucas Falch, Ms. Swapna Premasiri, and Mr. Hiroshan Gunawardane for their friendship and kind help. It has been a pleasant time working with you all.

I wish to express my sincere appreciation to Prof. Max Q.-H. Meng from the Department of Electrical Engineering at the Chinese University of Hong Kong (CUHK), Hong Kong. I am thankful to him for offering me the visiting opportunity at the CUHK. It was a particularly rewarding experience. I have benefited a lot from his insightful perspectives. I have also learned a lot from the members in his group. Special thanks go to Mr. Chaoqun Wang for his help and the countless beneficial discussions that we have had during my visit to CUHK.

I am also thankful to my research project sponsor. This work has been supported by research grant grant held by Dr. Clarence W. de Silva through the India-Canada Centre of Excellence for Innovative Multidisciplinary Partnership to accelerate Community Transformation and Sustainability (IC-IMPACTS).

Finally, I would like to express my deepest thankfulness and gratitude to my family. Thank you my parents for your patience, encouragement, and unconditional love. Without your support, this work would not have been possible.

Chapter 1: Introduction

This dissertation concerns robotic sampling and path planning for environmental field estimation and the practical issues in aquatic monitoring, particularly of surface water. Both quantitative and qualitative evaluations of the monitored environment are investigated in the present work.

1.1 Motivation

Monitoring programs of aquatic environments play a critical role in various uses of water such as the study of aquatic life, livestock water provision, human usage, agriculture and irrigation, and ecosystem conservation. Clean water sources are necessary not only for the aquatic ecosystem and natural habitats, but also for public health. In the past, water body evaluation has relied primarily on time-consuming and human-intensive field measurements for data collection. Technicians usually test water sources in the field utilizing hand-held devices, or transport water samples to laboratories for further analysis. The monitoring programs of this manner have been limited by their inadequate measurements on both spatial and temporal scales.

Recent advances in the technologies of sensing, robotics, and wireless sensor networks have led to significant progress in environmental telemonitoring. Automated systems that can carry out on-line sensing, wireless communication, and autonomous decision-making have been widely implemented in real-world applications. When compared with the traditional measuring processes, automation techniques can provide insitu sampling, automatic processing, and on-line early warning, and can be deployed in remote and hazardous environments.

In the field of aquatic monitoring, static stations or buoys with capabilities of automated measurement, data logging, and wireless transmission have been designed by research institutes [1], [2] or deployed by environmental agencies [3], [4]. Although automated systems with static sensing agents have been implemented to provide on-line measurements, their applications have been impeded by the inadequacy and inflexibility in spatial or spatiotemporal surveillance.

1

In contrast, robotic sensing systems can offer unprecedented flexibility, efficiency, and effectiveness in information gathering, which improve the performance of monitoring process, accommodate complex variations of environmental phenomena, and provide rapid response to unexpected events or anomalies. Consequently, robotic sensing systems have been developed and deployed to provide spatiotemporal measurements of water sources such as pools, lakes, reservoirs, rivers, and oceans. The main research focus in this area has been primarily on system design and development [5]–[8], sensor scheduling and path planning [9]–[13], environment modeling and field reconstruction [14]–[17], and data interpretation and evaluation [18].

1.2 Background and Challenges

In a mobile sensing process, robots travel to different locations for carrying out automated sampling and data acquisition. Depending on the monitoring goals, the robot is equipped with appropriate sensors to measure the necessary variables/parameters (physical, chemical, biological, radiological, and so on) [19]. A specific type of sensor may be able to acquire data from multiple geographical locations simultaneously. For example, a thermal camera can capture the temperature distribution over a large spatial area. Many other types of sensors, however, may only acquire data from a single location at one time. For example, a conductivity sensor measures the electrical conductivity at just one site at a time. These sensors are termed "point sensors" in the present work. The measurements from them can provide the information pertaining to the corresponding sampling locations only. Then, to obtain spatial information, the robots with point sensors have navigate to many sampling locations over the study area to take the necessary measurements. Afterwards, the data collected by the point sensors are analyzed to interpret the environmental field or predict the physical quantity at the unobserved locations.

Quantitative analysis of an environmental field with point sensors has been actively studied in the areas of geostatistics, robotics, wireless sensor networks, and environmental monitoring. To tackle a variety of analysis tasks such as exploration, estimation, prediction, and extremum seeking, it may be useful to integrate a spatial or spatiotemporal statistical model [20]. In the spatial statistics, the study area is generally modeled as a random field. Observations from the sampled locations are utilized to characterize the spatial profile,

estimate its underlying environmental model, predict the information at unobserved locations, or reconstruct the scalar field of the monitored area.

Sampling sites are designed by following a specified frame, which affects the estimation performance of the environmental field. It is evident that sampling at a higher resolution that covers the overall study field will provide better estimation results that are close to the ground truth of the underlying environmental phenomenon. However, mobile sensing robots in the field have limited on-board resources, especially limited energy storage and power supply (battery, fuel, etc.), which restrict the number of data samples obtained by the robots as well as the associated area of coverage (spatial point of view). On the other hand, these constraints may lead to insufficient data collection when the phenomena in the monitored field varies significantly (temporal point of view). The major challenge for robotic sensing applications resides on how to plan an effective and efficient sampling mission to guide resource-constrained robots to visit and measure more representative and bountiful sites over the monitored area.

In aquatic monitoring programs, qualitative analysis of a water body is based on its physical, chemical, and biological parameters. With the objective of providing an overall representation of the water quality based on all measurements, effort has gone into developing Water Quality Indices (WQIs) [21]. A WQI provides a convenient way to represent various and complex aspects that determine the water quality by aggregating the measured data of water quality parameters into a numerical score, and then categorizing them into several quality classes, for the purpose of reporting to the public, agencies, policy makers, and other authorities.

The implementation of mobile robotic sensing facilitates automated data acquisition from multiple robots by multiple sensors at high sampling rates in an effective manner. Large volumes of on-line data gathered in this manner can be rather complex and difficult for processing in evaluating the overall quality of water. This has led to the emergence of a formula that is expressed in the form of a WQI to encapsulate large quantities of water quality data into a simple score to categorize the water quality. However, in the traditional monitoring processes, the existing WQIs have been applied for long-term quality evaluation using data that are collected manually at low sampling rates. Indexing using online measurements has been rarely considered, and the particular types of WQI that are used have some shortcomings.

1.3 Research Objectives and Contributions

In recent years there have been increasingly stringent requirements in the combined use of wireless sensor networks and mobile sensing robots for on-line environmental monitoring. In the present dissertation, the primary research goal is the investigation and development of novel methods to facilitate the automated monitoring process of aquatic environment by using a wireless robotic sensing system through on-line measurements, real-time planning, and automatic information processing.

Specifically, rapid deployment is the first objective of implementing the robotic sensors. Therefore, on-line planning is required when generates sampling missions for sensor scheduling. Furthermore, adaptive sampling that can carry out a sensing process at more significant sites is expected to predict environment field properties. For quantitative estimation, the measurements at the sampling sites are utilized to construct an accurate field map over the entire study area. Meanwhile, a clear index that can interpret the quality of the monitored water source is needed for qualitative evaluation.

Consider limited samples by discrete sensors, constrained power supply, and limited computational capabilities, to predict a large-scale field with varying properties is a challenging issue. In addition, a mannerism that can integrate large volumes of on-line measurements to a simple interpretation of water quality is another crucial issue. With the objectives and issues, the main contributions of the dissertation are listed as follows:

• **Regular grid-based exploratory survey**: The first key contribution is the development of new robotic sampling and path planning schemes for a regular grid-based exploratory survey of an environment. First, two novel hexagonal grid-based survey planners are proposed to generate a sampling design that provides spatially balanced coverage and an energy-efficient sampling path cycle that can visit the planned sampling locations. The proposed planners are computationally feasible in polynomial time. As the related second contribution, the proposed survey planners are further extended to an energy-constrained scenario where the optimal coverage density for exploratory sampling is determined by a limited power supply budget.

The proposed sampling planners are applied for both the quantitative field estimation in a Gaussian Process (GP) scheme and the qualitative evaluation in water quality indexing.

- Informative path planning: A new Gaussian Markov Random Fields (GMRF) built-in hierarchical informative path planner is developed for robotic adaptive sampling. The main advantages of the proposed algorithm is the practical implementation of mobile sensing agents in spatiotemporal monitoring. First, the high-level planner designs the most informative sites in a global view over the entire spatial field, which is executed anytime for planning to navigate the robots continuously over time. Second, the low-level planner searches the informative locations locally while heading to the optimal sites that are assigned by the higher-level planner. The proposed framework provides an effective tradeoff between the computationally intensive global optimization for optimal experimental design and fast local greedy search for on-line path planning.
- **On-line surface water monitoring**: An on-line Wireless Mobile Sensor Network (WMSN) is designed and developed for on-line monitoring of surface water. Furthermore, a novel On-Line Water Quality Index (OLWQI) is developed to integrate and interpret the large volumes of on-line measurements acquired through automated sampling at high data resolutions.

The primary emphasis of the present work is on the two-dimensional spatial field analysis and its application in the monitoring of surface water of aquatic environments using mobile sensing robots such as Unmanned Surface Vehicles (USV) and Unmanned Aerial Vehicles (UAV). However, the developments in the present research can be implemented in mobile sensing system in any other environmental monitoring scenarios such as urban, atmospheric, marine, and land processes.

1.4 Organization of the Dissertation

The rest of the dissertation is organized as follows:

Chapter 2 surveys the literature and highlights the existing work related to the present work, including automated systems in environmental monitoring, sampling design and path planning, field estimation and mapping, and water quality evaluation. Chapter 3 addresses the problem of survey planning for distributing the sampling locations over a study area and generating the paths for the robots to visit and collect data from those locations. Given a target sampling resolution, the sampling locations of interest for the prior survey are selected based on a cellular decomposition that is composed of uniform hexagonal cells. They are visited by the robots along a path cycle that is created by the two proposed algorithms for coverage path planning. One generates the coverage path by circumnavigating the Minimum Spanning Tree (MST) that is constructed based on the proposed Auxiliary Coarse Cell (ACC) decomposition. To improve the computational efficiency, the other algorithm generates the coverage path cycle directly by rules conditioned on the ACCs. In this chapter, the proposed algorithms are compared with the state-of-the-art approach through experimentation.

Chapter 4 presents an energy-constrained survey planner for the exploration and mapping of an unknown environment. The proposed planning scheme generates the coverage paths for the robots subject to their energy constraints, targeting on environmental exploration and field mapping. The planned path results in an optimal sampling resolution, subject to the energy budget. The robot travels along the planned path cycle to collect data samples. Subsequently, the data is used to estimate a statistical model of the underlying environmental phenomenon as represented by a GP model. A robust Kriging method is utilized to estimate the GP structure and construct the scalar map of the monitored parameter. The experimental results are presented and compared with the state-of-the-art approaches while highlighting the superior performances of the proposed algorithm with regard to the estimation accuracy and the computational cost.

Chapter 5 addresses the issue of informative path planning and adaptive sensing in environmental field estimation with a mobile sensor network. A hierarchical planning framework is developed, which operates local greedy planning at each mobile sensing agent while determining the near-optimal sites at a sink by exploiting the global information over the entire spatial field. Mutual information (MI), an information-theoretic criterion, is utilized to quantify the significance of the sites in terms of their "informativeness." The proposed hierarchical framework provides a practical solution for on-line adaptive sampling and field mapping using a mobile sensor network. Chapter 6 presents the design and development of a WMSN for on-line monitoring of surface water in the field. For quality evaluation, an On-line Water Quality Index (OLWQI) is developed and implemented in the system to interpret vast quantities of online measurements. The index formulations are modified by the state-of-the-art index, CCME WQI, which has been developed by the Canadian Council of Ministers of Environment (CCME) for off-line indexing. Through experiments using a real-world dataset and a physical field test using the developed WMSN, it is shown that the proposed index is able to provide effective and reliable performance in online indexing of a large volume of measurements. In this chapter, the components of the developed WMSN are introduced in detail. The WMSN has been deployed in a real aquatic environment, and its performance is demonstrated.

Chapter 7 provides the conclusions pertaining to the research of the dissertation. Suggestions for possible future work and research topics are also indicated in this chapter.

Chapter 2: Related Work

2.1 Automated Environmental Monitoring

In the past decade, many automated platforms have been developed and implemented for remote sensing and environmental monitoring. Depending on the number of sensing agents in a system, the platforms can be categorized into two major types: systems with a single monitoring station (e.g., [3]) and systems with multiple sensor nodes in a network (e.g., [1], [5], [8], [22]). A single monitoring station commonly has sufficient power, computation resources, and communication resources. The main shortcoming of deploying a single station is the lack of ability to provide spatiotemporal monitoring over a large geographical area. A monitoring network with multiple sensing agents, in contrast, is capable of both spatial and temporal monitoring simultaneously.

Concerning the mobility of the agents, the sensor networks can be classified as static (e.g., [1], [4], [22]) and mobile (e.g., [5], [7], [8], [23], [24]). The static sensor networks have their sensing agents deployed at predetermined fixed locations, and provide continuous online measurements in the field. These systems have proven to be effective in supporting environmental monitoring in a timely manner due to their abilities of data requisition, information processing and wireless transmission [2]. However, they are hindered by their inadequacy and inflexibility in spatial sensing for area surveillance. In contrast, mobile sensor networks that consist of robotic sensing agents are able to carry out measurements by travelling over a large spatial area. These systems provide the capability of information gathering at variable locations of interest over the study area.

In aquatic monitoring applications, scientists in the fields of environmental science, biology, chemistry, geography, and robotics have increasing interests in leveraging advanced robotic sensing technologies to gather useful information of water sources. Many systems have been implemented with mobile sensing robots. For example, the design and initial application of a sensor-actuated network that consisted of multiple stationary buoys and one mobile robot have been introduced in [5] for sensing microbial communities in aquatic ecosystems. In [7], a low-cost, self-sustained mobile surface vehicle has been designed for water quality monitoring, which has been utilized as a part of a community science system to help in the environmental recovery of polluted bodies by performing on-

line operations. For autonomously sampling water columns, an energy-efficient and highly maneuverable gliding robotic fish has been developed in [23]. The hybrid system with underwater gliders and robotic fish has been used in field tests for sampling harmful algae concentration in a lake. A UAV-assisted water quality measurement system has been developed in [25] for in situ surface water quality measurement. The collected data was used to interpret the spatial distribution of temperature, conductivity, dissolved oxygen, and pH levels in a pond. With the support of robotic sensing, measurements of the objective parameters over both spatial and temporal scales can be obtained to estimate, interpret and reconstruct and environmental field of interest.

2.2 Field Sampling

In order to monitor natural resources, an environmental survey is required. In the survey, a systematic measurement process has to be carried out through a series of sampling missions at different sites, to observe variations of the environmental properties.

A distribution of the sampling locations provides a pattern across the study area, which indicates the estimation performance of the underlying random field. The distribution of the surveyed locations requires a Sampling Design (SD) that allows for the generation of a reliable interpretation of attributes of interest in the study area. The SD method determines the quality of the resulting survey. Typically, a suitable SD helps reduce errors in statistical inference [26].

2.2.1 Sampling Design

Sampling design strategies can be categorized as two main types: design-based sampling and model-based sampling. The former type incorporates probability sampling and design-based inference while the latter type conducts purposive sampling and spatial interpolation [27].

Various design-based techniques are available for selecting probabilistic samples. For instances, simple random sampling and systematic random sampling are straightforward probabilistic sampling techniques [28], which draw samples randomly and independently over the study area with equal probability. Stratified sampling divides the population into strata from which a sample is drawn by simple random sampling [29]. Latin hypercube sampling is a maximally stratified random sampling method, which guarantees the full coverage of multi-variate distribution in the feature space [30]. Design-based approaches focus on probabilistic sampling and inference for estimating quantities of environmental properties in terms of their totals, means, quantiles, standard deviations, and so on [31]. Design-based inference primarily seeks an unbiased estimator of a population of an environmental variable.

Purposive sampling selects representative sites for the model-based approaches to spatialize the surveyed field. Systematic sampling on a regular grid, also referred to as grid sampling, provides even geographical coverage by dividing the sensing domain into regular grids while selecting sampling sites at the grid nodes [32]. To fill the monitored space as uniformly as possible, spatial coverage sampling optimizes the distances between the nodes of an interpolation grid and the sampling sites. The distance measure, such as the Mean Squared Shortest Distance (MSSD), can be minimized by the k-means clustering algorithms [33] or the spatial simulated annealing algorithm [34]. In addition, geostatistical sampling optimizes the sample distribution by minimizing the (mean or maximum) variance of a model-based interpolation approach, such as Kriging [35], [36]. The optimal sampling pattern depends on the spatial correlation that can be depicted by a variogram, a correlation function, or a covariance function. The model-based approaches are particularly suitable for estimating an environmental model of an underlying field, predicting values at unobserved locations, and mapping spatial variation [37].

In a practical application, the choice between different sampling design techniques depends on the monitoring goal and the measuring system that is used to carry out the designed sampling process.

2.2.2 Exploratory Survey

To investigate an unknown environment, a preliminary sampling phase is required through a prior survey, which is called an exploratory survey phase [38]. Coverage sampling or robust sampling that is done to relax unrealistic prior assumptions is called an exploratory sampling design, which provides sufficient flexibility to explore an unknown process structure. Such designs have been adopted in spatial mapping without rigid prior assumptions, for initial deployment for prior knowledge collection, exploratory implementation for long-term design improvement, or combination with optimum designs for improved estimation performance [39]. In most existing monitoring programs, an initial deployment using exploratory sampling is required to explore and learn the underlying field.

Grid sampling accomplishes such requirements as exploratory designs [32], [40]. It distributes sampling locations by decomposing the sensing field into a grid of cells (e.g., squares, triangles, hexagons) where the sampling plots are taken in the grid tessellation. Spatial coverage sampling that optimizes the distance metric can also be implemented in an exploratory survey [33], [41]. In the absence of prior knowledge, a regular grid or an optimized spatial coverage design ensures overall coverage of the surveillance area. In particular, it provides a spatially balanced data sample distribution across the study area.

2.2.3 Coverage Path Planning for Sampling

Given the sampling locations of interest, the sampling paths for the mobile sensing robots to visit them can be directly planned by connecting all the target points. To tackle this problem, the Travelling Salesman Problem (TSP) [42] has been widely applied. It finds the shortest path by visiting each point exactly once, which leads to an energy-efficient route for mobile sensing. Orienteering Problem, an extended problem of TSP, has been studied to cover as many target points as possible while ensuring that the energy cost is within the given budget [43]. The TSP and its variations including the OP are Nondeterministic Polynomial-time hard (NP-hard) [44] methods; thus the computational cost is extremely high when the objective points are distributed over a large scale.

Rather than connecting the target points directly to form a sampling path, many planners generate a coverage path first and then obtain the sampling locations along the path. Coverage path planning (CPP) determines a path that covers the points in a target area as much as possible. It was originally utilized to fill the region with paths that are less overlapping and less repeating [45]. CPP has been extended to generate coverage paths for robotic sampling. In the literature, cellular decomposition methods such as Boustrophedon path [46], lawnmower path [18], and their variants [9], [47], have been used to generate the coverage path with a transect survey pattern consisting of a series of parallel linear transects to cover the decomposed cellular regions. Then data samples are taken while

traveling back and forth along the generated coverage path. The work in [45] reviews planning strategies that can create a coverage path.

2.3 Environmental Field Estimation

2.3.1 Random Fields

Environmental phenomena are reflected by complex interactions of physical, chemical and biological processes. The variation of an environmental property appears to be random, which is commonly modeled as a random field (or stochastic field) in spatial statistics. Field estimation and reconstruction through mobile sensing is an active research topic in the field of robotics, wireless sensor networks, geostatistics, and so on. For spatial data analysis, although model-free sampling approaches have been proven to be effective in applications such as stochastic source seeking [48], a spatial statistical model has attracted attention of environmental scientists and robotics researchers. Model-based approaches can estimate the underlying structure of the data generating process and provide the spatial and temporal characteristics and global information of the environment. They have been widely studied in pattern modeling [16], environmental mapping [49], information exploitation [50], boundary detection [47], and so on.

The sensed spatial field of interest is generally considered as a 2-dimensional plane $A \subset \mathbb{R}^2$. The continuous plane is discretized as a set of locations \tilde{S} with the possible sampling locations. A random field $Z = \{Z(\mathbf{s}_n), \mathbf{s}_n \in \tilde{S}\}$ is a set of random variables $Z(\mathbf{s}_n)$ indexed by its associated location \mathbf{s}_n , $n = 1, 2, ..., \mathcal{N}$. The spatial field $Y = \{Y(\mathbf{s}), \mathbf{s} \in \tilde{S}\}$ can be modelled as the combination of a deterministic term $X(\mathbf{s})$ and a random term $Z(\mathbf{s})$, that is:

$$Y(\mathbf{s}) = X(\mathbf{s}) + Z(\mathbf{s}). \tag{2.1}$$

The deterministic term $X(\mathbf{s})$ is a regression function that captures the trend over space. The random term $Z(\mathbf{s})$ is a random function that is specified by the finite-dimensional joint distribution of the random variables as:

$$F(z_1,...,z_{\mathcal{N}};\mathbf{s}_1,...,\mathbf{s}_{\mathcal{N}}) = p(Z(\mathbf{s}_1) \le z_1,...,Z(\mathbf{s}_{\mathcal{N}}) \le z_{\mathcal{N}}).$$
(2.2)

where z_n denotes the value of the random variable at the location \mathbf{s}_n , $n = 1, 2, ..., \mathcal{N}$.

In the model-based statistical data analysis, characteristics of a random field are interpreted by its stochastic structure. Stochastic process techniques have been investigated to describe complex environmental fields in space and time. Measurements are utilized as the observations to learn the underlying environmental model, interpret the statistical characteristics, and map the scalar spatiotemporal field. The performance of field reconstruction can be quantified by the mapping error, which is designated as performance metric.

2.3.2 Gaussian Processes

Effective statistical techniques have been taken an important role in the environmental analysis. Gaussian process (GP), a nonparametric Bayesian scheme, has been actively studied as a random field to model the underlying environment and provide estimation, prediction, and mapping of the study area. By generalizing a Gaussian distribution in a finite vector space, to a Gaussian process in a function space of infinite dimension, the GP framework can be used to model various physical phenomena and estimate values at any location in the sensing domain [20], [51].

In the context of the GP scheme, a Gaussian Random Field (GRF) is defined by any finite collection of random variables as:

$$\mathbf{Y} = (Y(\mathbf{s}_1), Y(\mathbf{s}_2), \dots, Y(\mathbf{s}_N))^T \sim \mathcal{GP}(\boldsymbol{\mu}(\mathbf{s}_n), \boldsymbol{\Sigma}(\mathbf{s}_i, \mathbf{s}_j)).$$
(2.3)

where the process is specified by the mean function $\mu(\mathbf{s}_n) = \text{Exp}(Y(\mathbf{s}_n))$, $n = 1, 2, ..., \mathcal{N}$, variance σ^2 , and the covariance function (or kernel function) $\Sigma(\mathbf{s}_i, \mathbf{s}_j)$ with the entry for the element $\Sigma_{ij} = \text{Cov}(\mathbf{s}_i, \mathbf{s}_j)$, $i = 1, 2, ..., \mathcal{N}$, $j = 1, 2, ..., \mathcal{N}$, $i \neq j$. The covariance function is symmetric positive-definite. For example, the Matérn class of covariance functions has been widely studied [52], which can be expressed as:

$$\Sigma(\mathbf{s}_i, \mathbf{s}_j) = \Sigma(h) = \frac{\sigma^2}{\Gamma(\nu) 2^{\nu-1}} (\kappa h)^{\nu} K_{\nu}(\kappa h), \qquad (2.4)$$

where $h = \|\mathbf{s}_i - \mathbf{s}_j\|$ denotes the Euclidean distance, κ denotes the spatial scale parameter, ν denotes the Matérn smoothness, and K_{ν} denotes the modified Bessel function.

2.3.3 Gaussian Process Regression

The Gaussian process regression (GPR), also known as Kriging, has been proposed as the best linear unbiased predictor (BLUP) for estimating and predicting the random scalar field under the Gaussian process scheme [53]. Under the GP scheme, the values and estimation uncertainties at unobserved points can be obtained conditioned on the historical observations by GPR. Consider the study area of interest $A \subset \mathbb{R}^2$, given a set of sampling locations $S = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\} \subset A$ with the corresponding observations $\mathbf{y}(\mathbf{s}) = [y(\mathbf{s}_1), y(\mathbf{s}_2), ..., y(\mathbf{s}_N)]^T$, the distribution of $Y(\mathbf{s}')$ at the unobserved point $\mathbf{s}' \in \tilde{S} \setminus S$ is also Gaussian. The conditional mean and variance at the unobserved point \mathbf{s}' can be obtained as:

$$\boldsymbol{\mu}(\mathbf{s}' \mid \mathbf{s}) = \boldsymbol{\mu}(\mathbf{s}') + \mathbf{c}(\mathbf{s}')^T \boldsymbol{\Sigma}^{-1}[\mathbf{y}(\mathbf{s}) - \boldsymbol{\mu}(\mathbf{s})], \qquad (2.5)$$

$$\boldsymbol{\Sigma}^{2}(\mathbf{s}' | \mathbf{s}) = \boldsymbol{\sigma}^{2} - \mathbf{c}(\mathbf{s}')^{T} \boldsymbol{\Sigma}^{-1} \mathbf{c}(\mathbf{s}'), \qquad (2.6)$$

where $\mathbf{c}(\mathbf{s}') = [c(\mathbf{s}', \mathbf{s}_1), c(\mathbf{s}', \mathbf{s}_2), ..., c(\mathbf{s}', \mathbf{s}_n)]^T$ defines the covariances between the point \mathbf{s}' and the points in *S*. In this case, the mean, variance and covariance functions are known a priori. The prediction model is known as simple Kriging. When the mean function is unknown, models that are more complex should be applied, such as ordinary kriging (OK) and universal kriging (UK) [54]. Furthermore, when all these functions are unknown, geostatistical Kriging model has been used to make predictions for the unobserved points. More details regarding Kriging models can be found in [55].

2.3.4 Gaussian Markov Random Fields

For factorizing the covariance functions and making predictions, the computation scale of the GPR is $\mathcal{O}(\mathcal{N}^3)$ with respect to the covariance function $\Sigma \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$. To promote computational efficiency, the GMRF approach has been introduced to approximate the GP model, which describes the spatial field by designating the Markovian properties into the random field [56].

For a GRF $Y = \{Y(\mathbf{s}), \mathbf{s} \in \tilde{S}\}$, let NE_i denote the set of neighbors to a sampling location \mathbf{s}_i . If the neighbors in NE_i to a sampling location \mathbf{s}_i satisfies that

 $p(Y(\mathbf{s}_i) | \mathbf{Y}(\tilde{S} \setminus \mathbf{s}_i)) = p(Y(\mathbf{s}_i) | \mathbf{Y}(NE_i))$, the spatial field follows the Markovian property and can be modeled as a GMRF. The GMRF is specified as:

$$\mathbf{Y} = (Y(\mathbf{s}_1), Y(\mathbf{s}_2), \dots, Y(\mathbf{s}_N))^T \sim \mathcal{GMRF}(\boldsymbol{\mu}(\mathbf{s}_n), \mathbf{Q}(\mathbf{s}_i, \mathbf{s}_j)), \qquad (2.7)$$

where $\mu(\mathbf{s}_n) = \operatorname{Exp}(Y(\mathbf{s}_n))$ is the mean function; **Q** is called the precision matrix and $\mathbf{Q}^{-1} = \boldsymbol{\Sigma}$. The elements in **Q** are non-zero only for neighbors and diagonal elements, that is, the precision matrix is sparse. The probability density function of **Y** is:

$$p(\mathbf{Y}) = (2\pi)^{-\mathcal{N}/2} \left| \mathbf{Q} \right|^{1/2} \exp\{-\frac{1}{2} (\mathbf{Y} - \boldsymbol{\mu})^T \mathbf{Q} (\mathbf{Y} - \boldsymbol{\mu})\}.$$
(2.8)

The GMRF model is specified by the precision matrix that is sparse and can be handled by a Stochastic Partial Differential Equation (SPDE) approach, which substantially improves the computing efficiency [57]. A triangular mesh is generally constructed in the study field to define the vertices and their neighbors, which discretizes the continuous domain to an indexed GMRF. To calculate the precision matrix, the SPDE approach can provide the explicit solution of the elements of **Q** by making use of the Matérn covariance functions. The precision matrix for two-dimensional domains can be obtained as:

$$\mathbf{Q} = \tau^2 (\kappa^4 \mathbf{C} + 2\kappa^2 \mathbf{G} + \mathbf{G} \mathbf{C}^{-1} \mathbf{G}).$$
(2.9)

The elements of **Q** are determined by the hyperparameter vector $\boldsymbol{\theta} = (\tau, \kappa)$. More details regarding the hyperparameters, matrix **C**, and matrix **G** in Equation 2.9 are referred to [57].

2.4 Adaptive Sampling

An exploratory survey with coverage sampling design distributes plots densely and evenly over the measured area. Such a survey process has been implemented as an initial deployment to explore an unknown environment. The parameters and hyperparameters of an environmental model can be learnt by utilizing the data collected from the prior survey.

With accumulated knowledge from continues deployment, the sampling sites that are more crucial in making predictions can be determined based on the established environmental model or the data-driven rules. In statistics, the selection of objective sampling sites is considered as an experimental design problem [58]. The experimental design techniques have been applied in many practical scenarios. For example, the problem of sensor selection addresses the activation of the most informative subset of sensors [59]. Similarly, the problem of sensor placement has been studied to find the most informative locations to place the sensors [60].

In the literature, experimental design theory has been studied for the development of optimality criteria. There are different evaluation approaches in which different criteria have been proposed, mainly focusing on optimizing an information matrix or a gain that is determined by the established spatial statistical model. "Alphabetical" optimality approaches have been proposed using the properties of the corresponding information matrices [61]. In an effort to determine the optimal design in terms of uncertainty, there is rich literature on information-theoretic criteria. Conditional entropy has been widely chosen as a metric to evaluate the most informative site, which represents the quantity of uncertainty that remains once the other locations are sampled. For the observed locations in S and the unobserved locations in $\tilde{S} \setminus S$, the conditional entropy is expressed as follows:

$$H(Y(\tilde{S} \setminus S) | Y(S)) = -\int p(y(\tilde{S} \setminus S), y(S)) \log p(y(\tilde{S} \setminus S) | y(S)) dy(\tilde{S} \setminus S) dy(S).$$
(2.10)

where Y(S) and $Y(\tilde{S} \setminus S)$ represent the sets of random variables of the observed and unobserved locations, respectively; and y(S) and $y(\tilde{S} \setminus S)$ represent the possible values of the random variables of the observed and unobserved locations, respectively. The associated optimal design of \mathcal{N} locations can be obtained, which leads to the minimum conditional entropy as:

$$S^* = \underset{|S|=\mathcal{N}}{\arg\min} H(Y(\tilde{S} \setminus S) | Y(S)).$$
(2.11)

In addition, mutual information (MI) criterion was also developed as an informationtheoretic criteria to quantify the information gain of a sampled site [62]. In the context of the MI metric, the main goal is to find the optimal locations that can maximally reduce the entropy over the unobserved locations. The MI is expressed as follows:

$$MI(Y(S), Y(\tilde{S} \setminus S)) = H(Y(\tilde{S} \setminus S)) - H(Y(\tilde{S} \setminus S) | Y(S)).$$
(2.12)

The associated optimal design of \mathcal{N} locations can be obtained, which seeks the set that maximizes the MI as:

$$S^* = \underset{|S|=\mathcal{N}}{\operatorname{arg\,max}} MI(Y(S), Y(\tilde{S} \setminus S)).$$
(2.13)

2.5 Quality Indexing

Complex statistical summaries variable by variable and water body by water body have been used to describe water quality in traditional approaches, which hardly provide a clear picture of the overall quality of the water [63]. A WQI was first presented in [64]. Since then, the significance of using a WQI to indicate the quality of water in various sources has been recognized [65], and research has gone into further development of WQIs.

The WQIs that are currently available differ according to their selection of the representative water quality parameters and the respective regulatory objectives (guidelines), the index formulation for data aggregation, and the category interpretation based on the index score. Among these considerations, the index formulation is perhaps the most significant one as it presents the mannerism of the statistical integration from complex data to a simple score. WQIs have been proposed according to different formulation models, such as additive model - National Sanitation Foundation WQI [66], multiplicative model [67], harmonic model - Oregon WQI [68], and so on. An entirely different formulation model has been adopted by the Canadian Council of Ministers of Environment (CCME) [69]–[71], which is knowns as the CCME WQI. The index formulation of the CCME WQI incorporates three statistical factors: Scope, Frequency and Amplitude, by comparing tests of water quality parameters and their objectives. With its strength as a parameter-extensible and objective-oriented index, CCME WQI has been applied by many agencies and institutes throughout the world [72]–[75].
Chapter 3: Grid-based Survey Planning

This chapter addresses the problem of grid-based exploratory survey planning that involves coverage sampling design and associated path planning for multiple mobile sensing robots. Two hexagonal grid-based planning algorithms are investigated, to generate the coverage path cycle for the robots to visit sampling locations of interest in a spatially balanced manner. The performance of the algorithms is evaluated and compared with the start-of-the-art approach. The results demonstrate the reliability and efficiency of the proposed hexagonal grid-based survey planners in mobile sensor scheduling for environment surveillance.

3.1 Overview

For interpreting the qualitative or quantitative profile of a surveyed area in terms of environmental parameters, the selection of the sampling locations for data collection is a key consideration in carrying out the measurement process across the study area. To characterize the entire study area, especially for an unknown area without any prior knowledge, an exploratory survey is required as an initial deployment.

Coverage sampling that can distribute data samples rather evenly over the space is implemented to carry out the prior survey. A sampling frame based on a regular grid has been widely used in the coverage sampling design for field estimation [31]. The grid-based sampling frame decomposes the sensing domain into a grid of cells, to distribute uniform plots across the area of interest. Sampling sites of interest can be easily positioned in the filed as regular grid nodes or centroids. The design procedure of the grid sampling is simple and efficient. More importantly, no prior knowledge is required to complete the design. It is noted that a design where the sites are chosen as the grid nodes of an equilateral triangular grid (i.e., the grid centroids of an equilateral hexagonal grid) is more efficient than other regular grid-based sampling frames [32]. This is because of the maximum distance between the observed locations and the observed sites is the shortest one. The literature on the grid sampling for an exploratory survey is rather rich. However, many research activities, especially in geostatistics and environmental sciences, only focused on the design of coverage sampling frame, without considering the associated sampling path to visit and measure the generated sampling locations.

For a mobile sensing process, after establishing the sampling targets, paths have to be determined for navigating the robots to visit them for data collection. The TSP-based planners have been applied for generating the paths to visit the objective locations [42]–[44]. However, the NP-hard characteristics for solving a TSP-based planning algorithm have led to severe computational burden, which limits its application in real-world situations. How to generate an effective and efficient path to guide mobile sensors takes a crucial role in robotic sensing applications.

In the present chapter, the problem of the hexagonal grid-based survey planning is investigated. The primary focus of the present chapter is to develop a mission planner for a hexagonal grid-based exploratory survey using mobile sensing robots. The survey planner should incorporate the selection of sites for coverage sampling and the generation of the corresponding sampling path to visit them.

3.2 Coverage Sampling Design

The monitored environmental field is generally treated as a continuous planar area. The estimation of the field characteristics is interpreted based on the distribution of the sampling locations across the study area. In the present work, the Sampling Locations of Interest (SLoIs) are generated by utilizing a hexagonal cell decomposition approach to distribute the plots evenly across the monitored area. This sampling framework introduces spatially balanced sampling locations where the distances between any neighboring SLoIs are equal.

Let the set $A \subset \mathbb{R}^2$ represent the region of interest with its contour \hat{A} . Let the set $S = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\} \subset A$ represent \mathcal{N} SLoIs to be measured for data collection, where the element $\mathbf{s}_i \in S$, $i = 1, 2, ..., \mathcal{N}$, is a 2-dimensional coordinate vector of the surveyed field, i.e., $\mathbf{s}_i = (s_1, s_2)$. All sample locations are generated using a hexagonal grid-based decomposition of a continuous planar area. The cells that have their centroids within the study area are designated as the Sub-Regions of Interest (SRoIs). These cell centroids are

the SLoIs for collecting data samples. Let SL represent the set of SLoIs, and |SL| represent the number of SLoIs in the set.

Specifically, the study area A is first decomposed into a grid of cells by a hexagonal tessellation. The center of each hexagonal cell is chosen as a SLoI $\mathbf{s} \in S$ if it is located within the contour \hat{A} of the study area. An example is shown in Figure 3.1, where the thick red line denotes the contour \hat{A} , the hexagons with solid black contours denote the generated SRoIs, and the blue star markers denote the generated SLoIs. Accordingly, the sampling locations are created uniformly across the study area, spaced at $d = \sqrt{3} \cdot l$, where d is the distance between two neighboring SLoIs, indicating the sampling resolution of the survey, and l is the edge length of a hexagonal cell. After obtaining the locations for sampling, a path is required to visit these targets. In the following section, path planning based on the hexagonal tessellation is studied to construct the sampling path for the robot to travel through the generated SLoIs.



Figure 3.1: Hexagonal tessellation and design of coverage sampling with SLoIs.

3.3 Hexagonal Grid-based Survey Planning

In the designed coverage sampling frame, the sensing domain is decomposed into SRoIs with spatially-balanced SLoIs that are distributed over the entire study area. The present section investigates the hexagonal grid-based strategies for generating a coverage path that can visit the target locations.

3.3.1 Spanning Tree-based Coverage Survey Planner

The SLoIs are generated by following the sampling frame introduced above. To measure these objective points, a novel spanning tree-based path planning approach is first presented under the hexagonal cellular decomposition. This approach has been originally implemented on a coverage path planning problem using square cellular decomposition [76]–[78]. The proposed planner implements the spanning tree-based coverage path planning strategies drawn from these work on a novel coverage sampling problem together with a different cellular tessellation. The basic idea is to construct a Minimum Spanning Tree (MST) and subsequently generate a path based on the obtained MST to visit the sampling locations.

Let the set $T_{\min} = \{V_T, E_T\}$ denote the MST, where V_T and E_T are the sets of vertices and edges of the tree, respectively. To determine the vertices $v_T \in V_T$ for constructing the MST, an Auxiliary Coarse Cell (ACC) decomposition method is proposed. In this decomposition, each coarse cell contains four neighboring regular hexagonal polygons (fine cells). An example of the ACC decomposition is shown in Figure 3.2, where the polygons with thick black edges denote the coarse cells. The tree vertices are created by considering the number and the positions of the SLoIs within a coarse cell. The bottom left, bottom right, top left, and top right regular hexagonal cells (fine cells) within a coarse cell are labeled as fine-cells 1, 2, 3, and 4, respectively. The vertex creation strategies are summarized in Table 3.1. For a coarse cell with only one sampling location inside, no vertex is created. After creating the vertices $v_T \in V_T$ by traversing all the coarse cells, an MST $T_{\min} = \{V_T, E_T\}$ is constructed based on these vertices. Prime's or Kruskal's algorithm is generally used to span an MST. An example of the constructed MST using Kruskal's spanning tree algorithm based on the created vertices is shown in Figure 3.2. The green solid circles denote the created MST vertices, and the green lines between the MST vertices are the constructed MST edges.

Table 3.1: MST Vertex Creation.

Coarse Cell	Condition & Strategy		
	SLoIs exist in 1, 2, 3, 4 fine-cells. A vertex is created at the midpoint of the edge between 2 and 3 fine-cells.		
	Left (Right) figure: SLoIs exist in 1, 3 (2, 4) fine- cells. A vertex is created at the left (right) side of the coarse cell if its left (right) neighboring coarse cell has four SLoIs.		
	Left (Right) figure: SLoIs exist in 1, 2 (3, 4) fine- cells. A vertex is created at the bottom (top) side of the coarse cell if its bottom (top) neighboring coarse cell has four SLoIs.		
	Left (Right) figure: SLoIs exist in 1, 2, 3 (2, 3, 4) fine- cells. A vertex is created with the same strategies by considering this coarse cell as a 1, 2 and 1, 3 (2, 4 and 3, 4) coarse cell. If two vertices are created, select a random one and remove the other one.		
	Left (Right) figure: SLoIs exist in 1, 2, 4 (1, 3, 4) fine- cells. A vertex is created with the same strategies by considering this coarse cell as a 1, 2 and 2, 4 (1, 3 and 3, 4) coarse cell. If two vertices are created, select a random one and remove the other.		



Figure 3.2: Coarse cell tessellation and relative constructed MST.

To generate a path that covers the points in the sampling design, an ordered sequence that indicates the sequential SLoIs to be visited is generated. Thus a path is created to visit each SLoI in the order of the sequence. This task is carried out by finding the effective edges (path segments) between the neighboring SLoI pairs to form the final path cycle. A sequence of $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$ is planned to visit sampling locations $\mathbf{s} \in S$ based on the constructed spanning tree. Here |p| denotes the total length of the path, $SP = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$ represents the set of SLoIs to be visited along the path, and |SP| denotes the number of sampling locations in the set. A robot travels through the SLoIs by following the sequence order $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$.

The path starts at a predetermined starting location \mathbf{s}_1 , then visits the SLoIs by circumnavigating the constructed MST clockwise or counter-clockwise, and finally returns to the starting location \mathbf{s}_1 , forming a path ring. Specifically, starting from \mathbf{s}_1 , the planner iteratively finds the next target location (the next neighboring location to visit) according to the current location, and generates a path segment between the current location and the target location until it reaches \mathbf{s}_1 again. In each iteration, the fine-cell position of the current location inside its coarse cell is first identified. One of the four possible positions (fine-cell 1, 2, 3, or 4 in its coarse cell, see the star markers in Figure 3.3) can be identified for a given current location. The rules to find the next target location are based on the conditions of its surrounding MST vertices or edges. In Figure 3.3, the green solid circles and lines denote the possible MST vertices and tree edges surrounding the central coarse cell with respect to the four possible fine-cell positions. The proposed path planner with a counter-clockwise circumnavigation is given in Algorithm 3.1 with pseudo code.



Figure 3.3: Possible MST vertices and edges surrounding a coarse cell.

Algorithm 3.1: *circumnavigationPathGeneration*.

Input: $S, T_{min} = \{V_T, E_T\}, \mathbf{s}_1 \in S$ Output: $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N), \mathbf{s}_i \in S, i = 1, 2, ..., N$. 1 $v_{current} = \mathbf{s}_1$ 2 do 3 $v_C = getCoarseCellLocation (v_{current});$

4 switch getCellPosition $(v_{current})$ do

5	case 1 do			
6	if $e_{v_{B1},v_{BL}} \in E_T$ then $v_{next} = moveTo$ ('Left');			
7	else if $e_{v_{C},v_{B1}} \in E_T \parallel e_{v_{C},v_{B2}} \in E_T \parallel e_{v_{B1},v_{B2}} \in E_T$ then $v_{next} = moveTo$ ('Bottom Right');			
8	else if $v_C \in V_T$ then $v_{next} = moveTo$ ('Right');			
9	else $v_{next} = moveTo$ ('Top Right');			
10	case 2 do			
11	if $e_{v_{R1},v_{BR}} \in E_T$ then $v_{next} = moveTo$ ('Bottom Right');			
12	else if $e_{v_{C},v_{R1}} \in E_{T} e_{v_{C},v_{R2}} \in E_{T} e_{v_{R1},v_{R2}} \in E_{T}$ then $v_{next} = moveTo$ ('Right');			
13	else if $v_C \in V_T$ then $v_{next} = moveTo$ ('Top Right');			
14	else $v_{next} = moveTo$ ('Left');			
15	case 3 do $e_{v_{C},v_{L1}} \in E_T e_{v_{C},v_{L2}} \in E_T e_{v_{L1},v_{L2}} \in E_T$			
16	if $e_{v_{L1},v_{TL}} \in E_T$ then $v_{next} = moveTo$ ('Top Left');			
17	else if $e_{v_{C},v_{L1}} \in E_{T} e_{v_{C},v_{L2}} \in E_{T} e_{v_{L1},v_{L2}} \in E_{T}$ then $v_{next} = moveTo$ ('Left');			
18	else if $v_C \in V_T$ then $v_{next} = moveTo$ ('Bottom Left');			
19	else $v_{next} = moveTo$ ('Right');			
20	case 4 do			
21	if $e_{v_{T1},v_{TR}} \in E_T$ then $v_{next} = moveTo$ ('Right');			
22	else if $e_{v_{C},v_{T1}} \in E_T \parallel e_{v_{C},v_{T2}} \in E_T \parallel e_{v_{T1},v_{T2}} \in E_T$ then $v_{next} = moveTo$ ('Top Left');			
23	else if $v_c \in V_T$ then $v_{next} = moveTo$ ('Left');			
24	else $v_{next} = moveTo$ ('Bottom Left');			
25 v	$v_{current} = v_{next};$			
26 while $v_{current} = \mathbf{s}_1$;				

An execution example of the Algorithm 3.1 is shown in Figure 3.4. A path ring (the thick blue lines between SLoIs) is generated to circumnavigate the MST and visit the SLoIs inside the fine-cells.



Figure 3.4: Generation of the circumnavigation path based on the MST.

In this figure, some SLoIs are not visited by the generated circumnavigation path. Let *SN* represent the set of the unvisited SLoIs. To cover these remaining SLoIs, several simple strategies are applied. For an unvisited SLoI, if there is a pair of neighboring SLoIs with a constructed edge between them, first add two new path segments connecting the current SLoI to the two neighbors, and then remove the path segment between those two neighbors (see Figure 3.5(a)). In addition, if there are four neighboring SLoIs with a "Z" shaped pattern of three constructed edges between them (see Figure 3.5(b)), then add and remove rules of path segment update are shown in the figure. These two strategies have been defined as the *V*- and *Z*- modification in the work of [9]. More details are found in the cited reference. Update the coverage path by iteratively checking and implementing these strategies on all unvisited SLoIs.

By applying these strategies of path segment update, the planned coverage path cycle of the example in Figure 3.4 is modified as shown in Figure 3.6. Finally, the overall coverage path by the proposed algorithm is shown in Figure 3.7. The proposed hexagonal grid-based spanning tree coverage (HGSTC) survey planner is summarized in Algorithm 3.2 with pseudo code.



Figure 3.5: Strategies of path segment generation.

(a) *V*-modification; (b) *Z*-modification.



Figure 3.6: Overall path cycle after path update.



Figure 3.7: Final planned path for coverage sampling.

Algorithm 3.2: *hexagonalGridbasedSpanningTreeCoverageSurveyPlanner*.

Input: \hat{A} , d, $\mathbf{s}_1 \in S$.

Output: $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$.

- $SLoIs = hexagonalBasedSamplingDesign(\hat{A}, r);$
- $V_T = MSTVertexCreation (SLoIs);$
- $E_T = MSTConstruction (V_T, 'Kruskal's'); % Construct the MST$
- $p, SP = counterClockwiseCircumnavigationPathGeneration(S, V_T, E_T, \mathbf{s}_1 \in S)$
- $SN \leftarrow SLoIs \setminus SP$ % Unvisited locations

do

- $updateCells = \emptyset$
- 8 foreach $i \in SN$ do
- *cells*, *pattern* = *findNearCellsInPatterns*(*i*);
- **if** \sim *isEmpty* (*cells*) **then**
- $updateCells \leftarrow updateCells \cup i;$
- $p \leftarrow pathSegmentGeneration (p, cells, pattern);$
- $SP \leftarrow SP \cup i;$
- $SN \leftarrow SN \setminus i$;
- **while** *isEmpty* (*updateCells*);

3.3.2 Hexagonal Grid-based Coverage Survey Planner

The spanning tree-based path planner generates a path cycle that can guide a robot to visit the SLoIs. However, it traverses the cells twice to generate the coverage path, covering all coarse cells to construct the MST first, revisiting all fine cells to circumnavigate the MST. To improve the efficiency of path forming, the strategies of the hexagonal grid-based path planning is further investigated now.

Note that the edge candidates between the adjacent SLoIs compose a triangular grid pattern (see Figure 3.8). In the Figure, the dashed lines between the neighboring SLoIs denote the edge candidates. The planning goal is to determine the effective edges among the candidates to form a path cycle that visits each location only once and returns to its starting position. This problem is similar to the formulation of a Hamiltonian cycle on a specified triangular grid graph [79].



Figure 3.8: Edge candidates of neighboring SLoI pairs.

The effective edges are determined by utilizing the proposed ACC decomposition method. Specifically, the proposed planner traverses the ACCs one by one to generate the path segments depending on the conditions (number and position) of the SLoIs in the current ACC and its surrounding ACCs. In a coarse cell, the bottom left, bottom right, top left, and top right hexagonal polygons are designated as the fine-cells 1, 2, 3, and 4, respectively. The current coarse cell is considered as a full cell if it contains four fine cells. The neighboring four fine cells in its right and bottom neighboring coarse cells are defined as the cells 5, 6 and cells 7, 8, respectively. The related SLoIs of an ACC for the path segment generation is shown in Figure 3.9. In the figure, the current coarse cell is highlighted by a solid bold contour. The SLoIs of the current and its neighboring coarse cells are labeled by their numbers. The dashed blue lines show the possible path segments to be generated. When traversing ACCs, for a selected coarse cell that is full, the segment paths are constructed by the following rules:

- If the left neighboring ACC is full, generate the edge $e_{3,4}$; otherwise, generate the edges $e_{1,3}$ and $e_{3,4}$.
- If the right neighboring ACC is full, generate the edges $e_{2,5}$ and $e_{4,6}$; otherwise, generate the edge $e_{2,4}$.
- If the bottom ACC is full and the current ACC row has not been connected to the bottom ACC row, generate the edges e_{1,7} and e_{2,8}, and remove the existing edge e_{7,8}; otherwise, generate the edge e_{1,2}.



Figure 3.9: Rules of path segment generation in an ACC.

After executing these procedures, a path cycle is generated to visit all the ACCs that are full. An example of an execution result is shown in Figure 3.10. There are still unvisited SLoIs within the area contour whose ACCs are not full. To visit them, the path segment update strategies presented in section 3.3.1 are applied. In addition, it is noted that the planned results may form more than one path cycle. Then, any two neighboring cycles can be combined to form one cycle by a simple edge adjustment (see Figure 3.11). The planned coverage path ring after incorporating unvisited locations is shown Figure 3.12.



Figure 3.10: Generated path segments to visit full ACCs.



Figure 3.11: Combination of neighboring cycles.



Figure 3.12: Generated coverage path after incorporating all unvisited SLoIs.

In the proposed scheme, by spatial partitioning of the sensing domain in a hexagonal tessellation, the generated coverage path forms a Hamiltonian cycle that can guide the robot to measure plots that are distributed across the monitored area in a spatially balanced manner. When compared with the work in [80], the proposed path generation scheme is more robust for dealing with a triangular grid graph. More specifically, the compared work gets the Hamiltonian cycle starting from the boundary cycles of a 2-connected and

polygonal triangular grid (from outside to inside). In contrast, the proposed method deals with the path cycle from inside to outside by using the proposed ACC decomposition and the path segment generation strategies. Consequently, there are no requirements on the characteristics of the input triangular grid graph.

The data samples are acquired by following the generated coverage path with equally spaced distribution. An execution example of the final planned coverage path for sampling is shown in Figure 3.13. The proposed hexagonal grid-based coverage (HGC) survey planner is summarized by the pseudo code in Algorithm 3.3. The code from line 1 through 21 generates the coverage path to visit all full ACCs. The code from line 22 to 32 deals with the unvisited ACCs. In this manner, the planning algorithm generates a coverage path cycle to travel among the expected sampling locations for sensing.



Figure 3.13: Final planned path for coverage sampling.

Algorithm 3.3: hexagonalGridbasedCoverageSurveyPlanner.

Input : \hat{A} , d , $\mathbf{s}_1 \in S$.					
Output : $p = (\mathbf{s}_1, \mathbf{s}_2,, \mathbf{s}_N)$.					
1 $colACC$, $rowACC = getACCHexagonalGrid$ (\hat{A} , d);					
2 for $i = 1$: <i>colACC</i> do					
3 for $j = 1$: <i>rowACC</i> do					
4 $SLoIs = getCurCellLocation(i, j)$					
5 if <i>isCurrentACCFull</i> (<i>SLoIs</i>) then					
6 if \sim <i>isNeighbourACCFull</i> ('Left') then					
7 $p \leftarrow genPath(e_{1,3});$					
8 <i>flag</i> =1;					
9 if <i>isNeighbourACCFull</i> ('Right') then					
10 $p \leftarrow genPath(e_{2,5}, e_{4,6}, e_{3,4});$					
11 else					
12 $p \leftarrow genPath(e_{2,4}, e_{3,4});$					
13 if <i>isNeighbourACCFull</i> ('Bottom') && <i>flag</i> ==1 then					
14 $p \leftarrow genPath(e_{1,7}, e_{2,8});$					
15 $p \leftarrow removePath(e_{7,8});$					
16 $flag = 0;$					
17 else					
18 $p \leftarrow genPath(e_{1,2});$					
19 $SP = setFineCellVisited (SLoIs);$					
20 else					
21 $SN = setFineCellNotVisited (SLoIs);$					

if getPathCycleNumber(p) > 1 **then**

 $p \leftarrow pathCycleCombination(p);$

do

- $updateCells = \emptyset$
- 26 foreach $i \in SN$ do
- *cells*, *pattern* = *findNearCellsInPatterns*(*i*);
- **if** \sim *isEmpty* (*cells*) **then**
- $updateCells \leftarrow updateCells \cup i;$
- $p \leftarrow pathSegmentGeneration (p, cells, pattern);$
- $SP \leftarrow SP \cup i;$
- $SN \leftarrow SN \setminus i$;
- 33 while *isEmpty* (*updateCells*);

3.3.3 Sub-paths for Multiple Mobile Sensors

In a dynamic environment under natural phenomena, physical quantities of monitored parameters at a location may vary over time. Accordingly, the measurement at a sampling location should be done within a certain time interval. It may record the temporal characteristics of parameters under an objective sampling frequency or detect events within a required delay bound.

To satisfy the sampling frequency requirement for measuring at each sampling location, sub-paths along the generated path ring are assigned to multiple robots. Let the set $U = \{u_1, u_2, ..., u_M\}$ represent $\mathcal{M} \in \mathbb{Z}^+$ robots that are deployed in the monitored field. Notice that the data is collected at different locations along the planned path one by one in a time series. Let $t_{\mathbf{s}_i, \mathbf{s}_j}$ represent the time consumption when traveling from sampling location \mathbf{s}_i to \mathbf{s}_j . In order to avoid the water current disturbances induced by movement,

a robot stops and stays at an SLoI while carrying out the measurement process. Let t_M represent the time taken by the measuring process for staying at an SLoI. The period that all SLoIs are visited once is called a survey cycle or a sampling cycle.

If a single robot is deployed to follow the generated path ring $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$ periodically, the resulting time consumption is $t_p = |SP| \cdot t_M + \sum_{i=1}^{N-1} t_{\mathbf{s}_i, \mathbf{s}_{i+1}}$ for each sampling cycle. This means each SLoI is visited at every time interval t_p . To ensure that the sampling rate at each SLoI satisfies a given time interval (time budget) t_{udt} , if $t_p > t_{udt}$, the path ring p is divided into \mathcal{M} sub-paths, such that

$$\frac{|SP| \cdot t_M}{u} + \frac{|p|}{u \cdot v} \le t_{udt}, \qquad (3.1)$$

where *v* is the average speed of the robot. To find the minimum number of robots to be deployed, $\mathcal{M} = \left[(|SP| \cdot t_M) / t_{udt} + |p| / (v \cdot t_{udt}) \right]$ is obtained. The coverage path cycle *p* is uniformly divided into *u* sub-paths $p_1, p_2, ..., p_M$. They are assigned to a set of \mathcal{M} robots.

The generated path ring p starts and ends at the same location, forming a cycle route. Then sub-paths extracted from the path cycle are assigned to multiple robots to satisfy the time interval requirement for measurement at an SLoI. Thus, the mobile sensors take different sub-paths by moving along the path ring clockwise or counter-clockwise in different sensing cycles. In this scheme, each SLoI can be visited and sensed within the objective time interval t_{udt} .

3.4 Experimental Results

An exploratory survey using a regular grid is attractive in practical applications because of its simplicity and efficiency. The present work extends the sampling design by incorporating its path planning, which can benefit a rapidly deployable and on-line mobile sensing system.

The proposed planners have been tested in a USV-assisted sensing platform that was designed and developed in the Industrial Automation Lab of the University of British Columbia (UBC). The objective of the platform is to provide on-line monitoring of surface water. The platform has been deployed at the Yosef Wosk Reflecting Pool of the University of British Columbia, Canada. The aerial photograph of the study area is shown in Figure 3.14. The in situ deployment of a USV is shown in Figure 3.15. More details of the design and development of the system are provided in Chapter 6.



Figure 3.14: Aerial view of the Yosef Wosk Reflecting Pool.



Figure 3.15: Field deployment of a USV.

Before the system deployment, two presented survey planners were evaluated to generate the survey missions for the USVs. In order to evaluate the performance of the proposed algorithms, they were operated with respect to different sampling resolutions. Meanwhile the results were compared with the state-of-the-art TSP-based survey planner on hexagonal grid (HGTSP), which was solved using integer linear programming. The experiments were executed using Matlab R2017a in a desk-top computer (PC) with a 4.00 GHz Intel Core i7-6700K CPU, 32 GB RAM. The experimental results are given in Table 3.2. Due to the limited space, some of the paths planned by the algorithms are shown in Figure 3.16.

d	l	SL	Visited SLoIs SP			Total Path Length p		
(m)	(m)						(m)	
			HGTSP	HGSTC	HGC	HGTSP	HGSTC	HGC
4.6	2.7	74	74	74	74	340.4	340.4	340.4
4.4	2.5	81	81	81	81	356.4	356.4	356.4
4.2	2.4	86	86	86	86	361.2	361.2	361.2
4.0	2.3	96	96	95	95	384.0	380.0	380.0
3.8	2.2	107	107	107	107	406.6	406.6	406.6
3.6	2.1	122	122	122	122	439.2	439.2	439.2
3.4	2.0	134	134	134	134	455.60	455.60	455.60
3.2	1.9	150	150	149	149	480.0	476.8	476.8
3.0	1.7	170	170	170	170	510.0	510.0	510.0
2.8	1.6	198	198	198	198	554.4	554.4	554.4
2.6	1.5	226	226	226	226	587.6	587.6	587.6
2.4	1.4	266	266	266	266	638.4	638.4	638.4
2.2	1.3	316	316	316	316	695.2	695.2	695.2
2.0	1.2	383	383	383	383	766	766	766

Table 3.2: Algorithm performance of survey planning.

Note that the solution of the TSP approach leads to the minimum path length for traversing all points of interest, which provides optimal performance on energy and time efficiency. In Table 3.2, the two proposed HGSTC and HGC algorithms and the HGTSP algorithm guaranteed optimal path length when all SLoIs were visited. For some cases where unvisited SLoIs existed, the total path length was still optimal for covering all the visited SLoIs. The low number of unvisited SLoIs was because they had only one neighboring SLoI, and hence no path segments could be generated for them (see top left unvisited SLoI in Figure 3.16(b)).

The corresponding processing time of the compared approaches is given in Figure 3.17. The proposed hexagonal grid-based sampling design checks each fine cell to examine if it is in the irregular-shaped study area, which can be obtained in polynomial time as $\mathcal{O}(\mathcal{C})$ [81], where \mathcal{C} denotes the number of hexagonal grid centroids. The computational complexity of each approach determines its performance in terms of the algorithm processing time. As mentioned, the TSP-based solver is NP hard and the computational cost is extremely high when the number of sampling sites is large. In comparison, the HGSTC algorithm first traverses the coarse cells according to the ACC decomposition with the computational complexity of $\mathcal{O}(|SL|)$, construct the MST with $\mathcal{O}(|E_{\tau}| \cdot \log |E_{\tau}|)$, circumnavigates the MST with $\mathcal{O}(|SL|)$, and finally updates the possible unvisited samples with the worst-case complexity of $\mathcal{O}(|SN| \cdot \log |SN|)$. The HGC survey planner first constructs the path segments while traversing the coarse cells with the computational complexity of $\mathcal{O}(|SL|)$, and then updates the possible unvisited samples to get the final path with the worst-case complexity of $\mathcal{O}(|SN| \cdot \log |SN|)$. The performance with respect to the algorithm processing time was significantly decreased by using the proposed approaches. Specifically, the proposed HGC algorithm shows superior performance in computational efficiency.



Figure 3.16: Generated survey plans with respect to different sampling resolutions. (a) d = 4.2 m, l = 2.4; (b) d = 4.0 m, l = 2.3 m; (c) d = 3.0 m, l = 1.7 m.



Figure 3.17: Algorithm performance of processing time.

N/A: Computation did not complete within the time limit of 100 minutes. The time cost is the average time consumption for 10 executions.

In the field experiment, the USVs were deployed in a distributed manner throughout the monitored field. The HGC algorithm was chosen to generate the survey mission for the USVs. It was executed at the remote server and sent to the USVs via the base station. The USVs followed the received survey missions to collect the measurements at the scheduled sampling locations and transmitted them to the base station and then to the server through wireless communication.

The fully charged battery enabled the USVs to move continuously for about 80 minutes at the average speed v of 0.4 m/s. The total travel distance of a fully charged USV is about 1920 m. In the field test, the sampling resolution was set as 4.2 m. The time cost t_M for the measurement process at each sampling location was set to 10 seconds. The updating time interval t_{udt} was set to 15 min. Given these specifications, the results of the survey mission were obtained by |SL| = 86, |SP| = 86, |p| = 361.2 m, $\mathcal{M} = 2$, where 86 sampling locations were covered by the generated paths for USVs. Accordingly, two USVs were required in the field to carry out the survey mission and meet the time interval requirement $t_{udt} = 15$ min. The theoretical time interval for data collection at each sampling

location is estimated as $t_p = 10 \cdot \lceil (86/2) \rceil + (4.2/0.4) \cdot (\lceil (86/2) \rceil) = 881.5$ seconds $< t_{udt}$. In real-world applications, the travel time between two sampling locations may vary depending on the hydrodynamics of the environment. The worst-case time cost can be used if the robots do not follow the planned path properly.

In the field test, two USVs followed the planned sampling path for four times (i.e., four sensing cycles) while each sampling location was measured eight times over 2 hours during the test. The measurements were utilized to demonstrate the quality of the monitored water by indexing. The planned sampling path and the operated sampling mission are illustrated in the Figure 3.18. In Figure 3.18(a), the solid circle and the hollow circle on the path denote the division points of the sub-paths for the two robots. Figure 3.18(b) shows an operated sampling mission of a sensing cycle that is carried out by the two robots. The blue stars denote the sampled sites. The dash line and the solid line denote the travelled paths of the two robots, respectively. More details of the quality index problem are presented in Chapter 6.



Figure 3.18: Evaluation results of the survey mission.

(a) Planned sampling mission; (b) Operated sampling mission by the two USVs in the field.

3.5 Summary

This chapter studied the problem of exploratory survey using robotic sensing, involving coverage sampling design, coverage path planning, and mobile sensor scheduling. The coverage sampling frame was designed under a hexagonal grid-based framework such that the sampling distribution could be balanced over the spatial field. The proposed survey planning algorithms generated energy-efficient coverage paths for robots to visit the sampling locations of interest, which were derived in a computationally feasible manner of polynomial time. Multiple robots were scheduled to travel along the planned path such that each objective sampling location could be visited within the required time interval. The experimental results on path planning demonstrated the effective and efficient performance of the proposed survey planers in comparison to the state-of-the-art TSP-based algorithm for generating the coverage path for sampling.

The proposed hexagonal grid-based survey planning algorithms were designed for the application scenario in automated aquatic environmental monitoring. However, they can benefit in various other applications with the goal of short-term but high-resolution monitoring, such as initial deployment in any unknown spatial field, on-line tracking of complex spatial anomalies, analysis of micro-ecosystem variations, and so on.

Chapter 4: Energy-constrained Exploratory Sampling

This chapter presents a survey planner for energy-constrained mobile robots to effectively explore and map an unknown environment that is modeled as a random filed. The proposed planner generates an energy-efficient coverage path for robotic sampling with an optimal density of coverage sampling and the associated cost subjected to a power supply constraint. It reduces the computational complexity and the processing time when compared with the existing state-of-the-art algorithms. A spatial statistical model is used to represent the underlying environment as a random field. The model is established by a Gaussian Process (GP) with a spatial trend, which is estimated by the data that is collected by following the proposed survey planner. The performance of the developed approach is evaluated by numerical experiments using a real-world experimental dataset that is obtained from an aquatic environmental monitoring program. The experimental results illustrate the reliability and efficiency of the present work in exploring and mapping an unknown environmental field using energy-constrained mobile robotic sensors.

4.1 Overview

When interpreting quantitative environmental phenomena over a field of interest, a statistical model is established to represent its spatial or spatiotemporal variation. For an unknown environment, the underlying environmental model is learnt by utilizing observations that are taken across the study area. The approximation of the environmental field includes the components of the model structure, such as model order, basis function, and hyperparameters. Deviations of the approximations from the real values will lead to model misspecification and unacceptable estimation performance. In other words, the field estimation results will not be robust against misspecification of the environmental model (e.g., [15]) may cause inferior estimation performance in a practical situation where spatial trends exist and are unknown.

It is evident that sampling at a higher resolution that covers the overall survey field will provide better estimation results that are close to the ground truth of the underlying environmental phenomenon. However, mobile sensing robots in the field have limited onboard resources, especially restricted energy/power capacity, which limit the number of data samples that can be obtained by the robots and the associated area of coverage. Given the limited resources, planning a mission for mobile robots with sensor-enabled sampling to visit and measure more sites over the study area according to a more effective schedule is an important problem in the field of environmental exploration and mapping.

TSP-based approaches have been implemented to generate a sampling path. However, the TSP and its variations are NP-hard methods; hence the computational cost is extremely high when the objective sampling locations are extensive. Most critically, these methods do no provide freedom to iteratively adjust the sampling pattern for better estimation performance while ensuring that the cost of the resultant path does not exceed the available power. Some planners generate the path by directly connecting the preplanned targets, while others guide robots to take data samples during navigation along the preplanned coverage path. For instance, the Boustrophedon path [46], lawnmower path [18], and their variants [9], [47] have been applied to generate the coverage path for tasks of sampling planning. Although these planners can generate coverage paths, most of them mainly focus on the problem of sweeping over the study area without analyzing the effect of the resulting sampling frame on spatial field mapping. Additionally, a constrained total travel length has not been addressed or emphasized in these existing path planners.

To the best of our knowledge, scheduling of a robotic sensor under its energy constraint in random field exploration and mapping has not been adequately investigated. The present work introduces a scheduling framework that integrates both optimal coverage sampling design and coverage path planning, which guides the mobile agent to take measurements for estimating and mapping an unknown environment. The proposed method can plan an energy-efficient coverage path with an optimal sampling density of coverage under a power supply budget. It can provide more representative data samples to estimate a GP structure and construct a scalar map of a monitored parameter. Compared to the existing methods, it can provide more effective and efficient sampling path planning for field estimation and mapping with respect to energy consumption, reconstruction error, and computational complexity.

4.2 Random Field and Environmental Model

In the present work, the physical process of the underlying environmental phenomenon is treated as a random field and modeled as a Gaussian process. The associated nomenclature of the environmental data model is given below.

The region of interest is denoted by the set A with its contour \hat{A} . Let $\mathbf{s} \in \mathbb{R}^2 = (s_1, s_2) \subset A$ denote a 2-dimensional coordinate vector of the surveyed field. Any set of \mathcal{N} sampling locations is denoted as $S = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_{\mathcal{N}}\}$. In the random field, a physical quantity is observed from a random variable $Y(\mathbf{s}_n)$, which is related to its sampled location, $\mathbf{s}_n \in S$, $n = 1, 2, ..., \mathcal{N}$. To study the infinite collections of random variables in the random field, it is required to take finite realizations of the random variables and make inferences on the basis of these data samples and the spatial statistic model. In the present work, the random variable is represented by the model:

$$Y(\mathbf{s}_n) = X(\mathbf{s}_n) + Z(\mathbf{s}_n), \qquad (4.1)$$

where, $X(\mathbf{s}_n)$ is a regression function that defines the mean value of the process (spatial trend over space); and $Z(\mathbf{s}_n)$ is a stochastic function that defines the random variation. The basic idea here is that the regression function represents the largest variance in the spatial data (i.e., the general spatial trend) while the stochastic term interpolates the residuals for random effects.

Consider the finite random variables at the locations s in the set S. The model in Equation (4.1) is vectorized as:

$$\mathbf{Y}(\mathbf{s}) = \mathbf{X}(\mathbf{s}) + \mathbf{Z}(\mathbf{s}), \qquad (4.2)$$

where $\mathbf{Y}(\mathbf{s}) = [Y(\mathbf{s}_1), Y(\mathbf{s}_2), ..., Y(\mathbf{s}_N)]^T$, $\mathbf{X}(\mathbf{s}) = [X(\mathbf{s}_1), X(\mathbf{s}_2), ..., X(\mathbf{s}_N)]^T$, and $\mathbf{Z}(\mathbf{s}) = [Z(\mathbf{s}_1), Z(\mathbf{s}_2), ..., Z(\mathbf{s}_N)]^T$ are random vectors. In the context of GP, the random vector $\mathbf{Z}(\mathbf{s})$ is designated as a spatial stochastic, Gaussian process with zero mean, variance σ^2 and covariance function $\mathbf{C}(\mathbf{s}, \mathbf{s}')$. As a result, the random vector $\mathbf{Y}(\mathbf{s})$ is also a GP, of mean defined by $\mathbf{X}(\mathbf{s})$ and stochastic residual defined by $\mathbf{Z}(\mathbf{s})$.

In the present work, Equation (4.2) represents the model of the underlying environmental field. In this model, the random vector $\mathbf{Y}(\mathbf{s})$ represents the random variables at the sampled locations in *S*, with their corresponding realizations, $\mathbf{y}(\mathbf{s}) = [y(\mathbf{s}_1), y(\mathbf{s}_2), ..., y(\mathbf{s}_n)]^T$, which denotes the set of observations at these locations.

4.3 Energy-Constrained Exploratory Sampling

Statistical characteristics of a random field are interpreted according to the distribution of sampling locations over the field. To determine the locations of interest for data acquisition, in Chapter 3, the hexagonal grid-based survey planner is used to generate the coverage sampling design and plan the associated coverage path for robotic sampling. Afterwards, the coverage path can be assigned to the robots to perform the sampling mission. The robots travel along the paths by following the sequential SLoIs and sensing at each location.

This chapter focuses on more specialized path planning issues, addressing the problem of energy-constrained mobile sensing in practice. To this end, the survey planners in the previous chapter are extended and constructed under the energy constraint. An execution example of the grid-based design for coverage sampling in a hexagonal tessellation is shown in Figure 4.1. The solid red line denotes the contour \hat{A} of the study area A. The hexagons with black solid contours denote the SRoIs. The black dots in the SRoIs denote the cell centroids, which are SLoIs.



Figure 4.1: Edge candidates of neighboring SLoI pairs for path planning.

In the coverage sampling design, l denotes the edge length of a hexagon in the hexagonal grid; and SL and |SL| denote the point set and the total number of the generated SLoIs, respectively. In the context of mobile sensing, let the sequence $p = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$ represent a planned path that is assigned to a sensing robot, where $\mathbf{s}_n \in SL$, n = 1, 2, ..., N denotes a sampling location that is visited within the path p. Here |p| denotes the total path length. The robot visits and operates the measuring process at the locations by following the order in the sequence. Let the set SP and |SP| represent the set and the number of sampling locations that are visited along this path, respectively. Let $d_{\mathbf{s}_i,\mathbf{s}_j}$ and $e(d_{\mathbf{s}_i,\mathbf{s}_j})$ represent the distance and energy cost when traveling from location \mathbf{s}_i to \mathbf{s}_j , respectively. In the measuring process, the robot stops at each sampling location to carry out the measurement. Let e_M represent the energy cost of traveling along the path p is expressed as:

$$e(p) = \sum_{i=1}^{N-1} e(d_{\mathbf{s}_{i},\mathbf{s}_{i+1}}) + e_{M} \cdot |SP|, \ \mathbf{s}_{n} \in SP \subseteq SL.$$
(4.3)

It should be ensured that the energy cost of the total path e(p) is within the given energy budge e_{bdt} .

The objective of the present planner is the generation of a sampling mission $p = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$ for scheduling a mobile sensor to collect data samples \mathbf{y} at the target locations *SL* with the densest sampling resolution under the power supply constraint e_{bdt} of the robot, which can be utilized to estimate the underlying environmental data model $Y(\mathbf{s})$ and map the scalar field A in a performance-oriented manner. This objective is formulated as

$$p^* = \arg\max_{p} |SP|, SP \subseteq SL,$$
s.t. $e(p) \le e_{bdt}.$
(4.4)

47

4.3.1 Optimal Density of Coverage Sampling

The distances between any neighboring SLoI pairs are equal. Given an edge length l of the fine cells in the hexagonal grid, the distance is given by $d = \sqrt{3} \cdot l$. With the introduced coverage sampling design, given the area contour \hat{A} , the total number of SLoIs to be surveyed, |SL|, can be determined as a dependent variable of l as well as d. In view of the equal distances between SLoI pairs, the total path length |p| of the generated coverage path p can be determined as $|p| = d \cdot |SP|$, $|SP| \le |SL|$. Given an energy budget e_{bdt} , finding the maximum point set SP, $SP \subseteq SL$, that the coverage path can achieve provides the most dense coverage and further improves the exploration performance of field estimation and mapping. The optimal sampling density d^* is determined by the optimal edge length l^* as:

$$l' = \arg \max_{l} |SP|, SP \subseteq SL,$$

s.t. $|SP| \le |SL|,$
 $e(p) = e(\sqrt{3} \cdot l) \cdot |SP| + e_M \cdot |SP| \le e_{bdt}.$ (4.5)

A decreasing grid spacing d may increase the resultant number of SLoIs, |SL|, inside the study area. However, there is no analytical expression for determining |SL| given l, since the shape of the study area is generally irregular and complex. The dependent variable |SL| has to be determined by executing the sampling design procedure for a given l. A brute-force search can be utilized to find the l^* that satisfies Equation (4.5) given a range of $l \in [l_{\min} : \Delta l : l_{\max}]$. The optimal $d^* = \sqrt{3} \cdot l^*$ is chosen as the optimal coverage density to generate the final coverage path. The efficiency of finding the optimal l^* is improved by using the binary search method to determine l^* such that $0 \le e_{bdt} - e(p) \le \delta$.

The ACC decomposition and the coverage path planning strategies guarantee that the total length of the generated path is bounded and indicative of the cell size. In view of this significant characteristic, there is no need to execute the path generation algorithm before determining the target edge length l. It considerably reduces the computational complexity and the corresponding processing time when finding the optimal coverage 48 density. After determining the optimum, the optimal sampling path can be obtained by applying the planning algorithms proposed in Chapter 3. The survey planner with optimal coverage density is summarized by the pseudo code in Algorithm 4.1.

Algorithm 4.1: *optHexagonalGridBasedSurveyPlanner*.

Input: $\hat{A}, e_{bdt}, \mathbf{s}_1 \in S$.

Output: $p = (\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N)$.

1 $l^* = getOptimalCoverageDensity(\hat{A}, e_{bdt}, searchMode);$

% searchMode = 'Binary' or 'Brute force'

2 *SLoIs* \leftarrow *genCoverageSamplingDesign*(\hat{A} , l^*);

3 $p \leftarrow hexagonalGridBasedSurveyPlanner(\hat{A}, SLoIs, \mathbf{s}_1, plannerMode);$

% *plannerMode* = 'HGSTC' or 'HGC'

4.3.2 Model Estimation and Field Mapping

The data samples taken along the path are used to estimate the underlying environmental model and build the field map. In the present work, the Universal Kriging (UK) method [55] is implemented to estimate the unknown field. The regression function $\mathbf{X}(\mathbf{s})$ of the environmental data model in Equation (4.2) is treated as a multivariate polynomial; that is:

$$\mathbf{X}(\mathbf{s}) = \sum_{i=1}^{\mathcal{I}} a_i f_i(\mathbf{s}) = \mathbf{f}(\mathbf{s})^T \mathbf{a}, \qquad (4.6)$$

here $\mathbf{f}(\mathbf{s}) = [f_1(\mathbf{s}), f_2(\mathbf{s}), ..., f_{\mathcal{I}}(\mathbf{s})]^T$ represents the basis function (e.g., the power base for a polynomial) and $\mathbf{a} = [a_1, a_2, ..., a_{\mathcal{I}}]^T$ represents the coefficients of the regression function.

In the context of GP, the random vector $\mathbf{Z}(\mathbf{s})$ in Equation (4.2) can be designated as a spatial stochastic GP with zero mean, variance σ^2 and covariance function $\mathbf{\Sigma}(\mathbf{s}, \mathbf{s}')$. The covariance function $\mathbf{\Sigma}(\mathbf{s}_i, \mathbf{s}_j) = \mathbf{\Sigma}(\mathbf{\theta})$, also known as a kernel function, describes the spatial dependence between locations \mathbf{s}_i and \mathbf{s}_j , $i \neq j = 1, 2, ..., \mathcal{N}$, where $\mathbf{\theta}$ denotes the hyperparameters. As a result, the random vector $\mathbf{Y}(\mathbf{s})$ can also be expressed as a GP, $\mathbf{Y}(\mathbf{s}) \sim \mathcal{GP}(\mathbf{X}(\mathbf{s}), \mathbf{\Sigma}(\mathbf{s}_i, \mathbf{s}_j))$, of mean defined by $\mathbf{X}(\mathbf{s})$ and stochastic residual defined by $\mathbf{Z}(\mathbf{s})$.

Given a set of samples $S = \{\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_N\}$ with the corresponding observations $\mathbf{y} = [y(\mathbf{s}_1), y(\mathbf{s}_2), ..., y(\mathbf{s}_N)]^T$, the prediction mean and variance are derived by utilizing the UK framework at locations $\mathbf{s}' \in \tilde{S} \setminus S$ as follows:

$$\hat{Y}(\mathbf{s}') = \mu(\mathbf{s}') = \mathbf{X}(\mathbf{s}')^T + \mathbf{c}(\mathbf{s}')^T \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{F}\mathbf{a}), \qquad (4.7)$$

$$\Sigma^{2}(\mathbf{s}') = \sigma^{2} - \mathbf{c}(\mathbf{s}')^{T} \Sigma^{-1} \mathbf{c}(\mathbf{s}') + \mathbf{M}^{T} (\mathbf{F}^{T} \Sigma^{-1} \mathbf{F})^{-1} \mathbf{M} , \qquad (4.8)$$

where $\mathbf{F} = [\mathbf{f}(\mathbf{s}_1)^T, \mathbf{f}(\mathbf{s}_2)^T, ..., \mathbf{f}(\mathbf{s}_N)^T]^T$ is the model matrix of attributes for the locations in *S*, $\mathbf{a} = (\mathbf{F}^T \boldsymbol{\Sigma}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \boldsymbol{\Sigma}^{-1} \mathbf{y}$ is derived through Generalized Least Squares (GLS), $\mathbf{c}(\mathbf{s}') = [c(\mathbf{s}', \mathbf{s}_1), c(\mathbf{s}', \mathbf{s}_2), ..., c(\mathbf{s}', \mathbf{s}_n)]^T$ defines the correlations between the point \mathbf{s}' and the locations in *S*, and $\mathbf{M} = \mathbf{f}(\mathbf{s}') - \mathbf{F}^T \boldsymbol{\Sigma}^{-1} \mathbf{c}(\mathbf{s}')$. The Kriging model is the Best Linear Unbiased Predictor (BLUP) for estimating and predicting the random scalar field under the Gaussian process scheme [55].

In the present work, with limited prior knowledge of the underlying field, the Blind Kriging (BK) approach [82] is adopted to identify its environmental model. The goal of BK is to efficiently designate the unknown basis functions by a Bayesian feature selection method in the regression analysis. In particular, it approximates the process $\mathbf{Y}(\mathbf{s})$ by extending the general Kriging model with additional candidate functions (features) in a linear model as:

$$\mathbf{X}(\mathbf{s}) = \sum_{i=1}^{\mathcal{I}} a_i f_i(\mathbf{s}) + \sum_{j=1}^{\mathcal{J}} b_j g_j(\mathbf{s}) = \mathbf{f}(\mathbf{s})^T \mathbf{a} + \mathbf{g}(\mathbf{s})^T \mathbf{b} , \qquad (4.9)$$

Here $\mathbf{g}(\mathbf{s}) = [g_1(\mathbf{s}), g_2(\mathbf{s}), ..., g_{\mathcal{J}}(\mathbf{s})]^T$ denotes the set of \mathcal{J} candidate functions and $\mathbf{b} = [b_1, b_2, ..., b_{\mathcal{J}}]^T$ denotes the corresponding scores of the candidate functions. Given the Gaussian process $Y(\mathbf{s})$, the corresponding scores \mathbf{b} are considered Gaussian, $\mathbf{b} \sim \mathcal{GP}(\mathbf{0}, \tau^2 \mathbf{R}(\mathbf{\Sigma}))$, where $\mathbf{R}(\mathbf{\Sigma})$ is defined as the variance-covariance matrix. The posterior estimation of \mathbf{b} can be derived by the BK estimation using the data samples \mathbf{y} .

More details of the BK method are found in [83]. In the present work, the Matérn 5/2 kernel, a member of the Matérn class in Equation (2.4), is selected as the covariance function, which is expressed as:

$$\Sigma(\mathbf{s}_i, \mathbf{s}_j) = \Sigma(h) = \sigma^2 \left(1 + \frac{\sqrt{5} \cdot h}{\rho} + \frac{5 \cdot h^2}{3 \cdot \rho^2}\right) \exp\left(-\frac{\sqrt{5} \cdot h}{\rho}\right).$$
(4.10)

The hyperparameters $\boldsymbol{\theta} = (\sigma, \rho)$ are identified through the Maximum Likelihood Estimation (MLE) [84] in the blind Kriging method.

It is noted that the random vector $\mathbf{Y}(\mathbf{s})$ is represented as 1-dimensional for the ease of notation. However, the generalized format for multi-dimensional outputs $\mathbf{Y}(\mathbf{s})$ is clear. In the next section, the performance of the proposed scheme is demonstrated through numerical experiments based on a real-world dataset.

4.4 Experiments and Discussion

The performance of the proposed HGC algorithm is evaluated using a real-world dataset. The simulation results are presented and discussed in this section. The proposed energyconstrained HGSTC and HGC algorithms are compared with the existing methods that focused on coverage path planning and sampling design, including Square Grid-based Spanning Tree Coverage (SGSTC) survey planner [85], Discrete Monotone Polygonal Partitioning-based (DMPP) survey planner [47], and Hexagonal Grid-based TSP (HGTSP) survey planner. The SGSTC approach was originally designed for sweep coverage planning. In the present work, it is adapted to deal with the coverage sampling problem in order to compare with the proposed sampling scheme. The DMPP approach, a variant of the traditional Boustrophedon method was designed to generate efficient coverage path for exploratory sampling. The HGTSP approach utilizes the same hexagonal grid-based sampling design in the present work but it plans the coverage path by a TSP solver.

All these planners can generate coverage paths at different resolutions to cover the surveyed field. To fairly compare the estimation performance on mapping, they are tuned to find the optimal spacing density that they can achieve for a specific mission power supply. Note that in these methods different parameters are used to adjust the coverage density of the sampling paths. In the SGSTC approach, the edge length of a fine square cell

is set to control the spacing. In the DMPP approach, the distance between the spaced transects determine the spacing density. The edge length l of a hexagonal grid defines the coverage density in the HGTSP, the HGSTC, and the HGC approaches. These parameters are defined as the Sampling Density Regulator (SDR) in the present simulation. Consequently, the Optimal SDR (OSDR) is the corresponding parameter for obtaining an optimal sampling density of coverage (i.e., l^* in Equation (4.5)).

4.4.1 Dataset Setup

This section presents the numerical simulation results of field estimation and mapping by implementing the proposed method and comparing with the state-of-the-art methods. In the simulation, a ground truth map over a surveillance area is chosen from a real-world dataset provided by the NOAA Operational Model Archive and Distribution System (NOMADS) [86]. The dataset records area measurements of surface salinity of the Caribbean Sea using a radiometer. In the dataset, an estuary area (latitude: 29.2410N to 30.2140N, longitude: 88.0933W to 89.1329W) with apparent spatial variation is chosen as the study area to evaluate the performance of the algorithm. A ground truth map is shown in Figure 4.2, where the colors indicate the surface salinity concentrations in units of psu. The GPS coordinates are translated into metric coordinates in units of km, as $\mathbf{s} = (s_1, s_2)$, $s_1 \in [0,51]$, $s_2 \in [0,56]$.



Figure 4.2: Ground truth map of the study area.

4.4.2 Estimation Performance

Given a power constraint, the coverage paths are planned with sampling points distributed on it. Data samples are acquired at the observed locations and then utilized to estimate the underlying field model and map the overall scalar field. The estimation performance is evaluated by comparing the prediction values with the ground truth values at the unobserved locations. The Root Mean Square Error (RMSE) is utilized as a measure to determine the mapping performance, which is defined as:

$$RMSE = \sqrt{\frac{1}{|I|} \sum_{\mathbf{i} \in I} \left[\hat{Y}(\mathbf{i}) | \mathbf{y} - y(\mathbf{i}) \right]^2} .$$
(4.11)

where I denotes the set of the grid nodes for spatial interpolation, $\mathbf{i} \in I \subset A$; |I| denotes the element number of the set; $y(\mathbf{i})$ denotes the ground truth value at a location $\mathbf{i} \in I$. A smaller RMSE indicates a better reconstruction result at the unobserved locations of the field. The Average Kriging Variance (AKV) is used as a measure to indicate the performance of estimation uncertainty, which is defined as:

$$AKV = \frac{1}{|I|} \sum_{\mathbf{i} \in I} \Sigma^2(\mathbf{i}) | \mathbf{y}.$$
(4.12)

The power supply budget determines the total distance a mobile robot can travel. For simplicity, the budget is designated in units of length rather than energy, in the simulation. To investigate the estimation performance with respect to the power constraint, different energy budgets are assigned to the proposed method and to the compared methods when planning the sampling paths. The results of the estimation performance are shown in Figure 4.3. Specifically, Figure 4.3(a) and (c) show the estimation results from a brute-force search to obtain the OSDR with the settings $l \in [2, \min(s_1, s_2)/4]$, $\Delta l = 0.1$. For comparison (see Figure 4.3(b) and (d)), the binary search approach is executed on these methods to obtain the OSDR with the settings $l \in [2, \min(s_1, s_2)/4]$. For brevity, some OSDR and mapping performance results with respect to the assigned energy budgets are presented in Table 4.1.


Figure 4.3: Algorithm performance in field estimation and mapping for different power supply settings.

(a) RMSE results using the brute-force search; (b) RMSE results using the binary search;(c) Results of AKV using the brute-force search; (d) Results of AKV using the binary search.

Energy Budget	Measure	SGSTC		DMPP		HGTSP		HGSTC		HGC	
e _{bdt}		Brute- force	Binary								
400	OSDR	4.7	5.36	5.5	5.52	N/A	4.74	4.6	4.58	4.6	4.58
	RMSE	2.8199	2.8272	2.3387	2.2149	N/A	1.7870	1.7576	1.7960	1.7576	1.7960
	AKV	1.1616	1.2134	1.4076	1.3072	N/A	1.3704	0.6920	0.7662	0.6920	0.7662
500	OSDR	4.5	4.51	4.8	4.84	N/A	3.78	3.7	3.68	3.7	3.68
	RMSE	2.5179	2.5179	1.8252	1.9065	N/A	2.1930	1.7120	1.5013	1.6752	1.4064
	AKV	1.2907	1.2907	1.4950	1.6925	N/A	1.9507	2.8755	1.4825	1.4622	0.9302
600	OSDR	3.8	4.05	4.2	4.16	N/A	3.10	3.1	3.10	3.1	3.10
	RMSE	1.5275	1.9496	1.7115	1.4642	N/A	1.1765	1.1909	1.1909	1.1765	1.1765
	AKV	2.8710	2.3832	1.2401	1.2136	N/A	0.5967	0.7232	0.7232	0.5967	0.5967
700	OSDR	3.6	3.76	3.7	3.68	N/A	N/A	2.9	2.91	2.9	2.91
	RMSE	1.4218	1.6517	1.7839	1.4064	N/A	N/A	1.1545	1.1473	1.1405	1.1358
	AKV	0.6660	0.8850	1.4368	1.0521	N/A	N/A	0.4000	0.3897	0.3012	0.2921
800	OSDR	3.3	3.25	3.4	3.35	N/A	N/A	2.5	2.51	2.5	2.51
	RMSE	1.6646	1.7153	1.5235	1.6199	N/A	N/A	0.9824	1.0384	0.9824	0.9214
	AKV	1.5737	0.7389	0.7997	0.6521	N/A	N/A	0.4704	0.5808	0.4704	0.4666

Table 4.1: Algorithm Performance of Field Estimation and Mapping.

N/A: denotes that the execution did not complete within the process time limit of 15 min.

As clear from Figure 4.3 and Table 4.1, the proposed HGC method provides more accurate results of field estimation and mapping when compared with the other approaches, as expressed by the RMSE and the AKV with different power supply constraints for both search tools. The HGTSP approach may provide the lowest RMSE in the limited number of experimental results but it is NP hard to solve. As a result, for the brute-force search or the binary search with a larger sampling density, the HGTSP approach is unable to find a plan within the limited time requirement, i.e., 15 minutes (see Figure 4.3 and Table 4.1). For the proposed two hexagonal grid-based methods, they generate similar estimation results by considering the RMSE results. The HGC method outperforms the HGSTC method in the AKV results. The difference between them is mainly on the computational complexity and is demonstrated in Section 4.4.3.

The results of sampling distributions, planned path, and their predicted scalar field of the proposed and compared methods with a budget of 500 are shown in Figures 4.4. The figures display information of the generated survey missions, the observed sampling locations, and the mapping results by making use of the observations at the sampled sites. The solid black dots and lines denote the planned SLoIs and the sampling paths, respectively. The colors indicate the corresponding predicted values at the unobserved locations in the study area. In the figures, it is shown that all the planners generated coverage paths for sampling across the study area. Meanwhile, the SGSTC, the HGTSP, and the proposed survey planners generated a coverage path cycle.

The results of the estimation uncertainty of the proposed and compared methods with a budget of 500 are shown in Figure 4.5. The colors in the figures display the corresponding estimated Kriging variances at the unobserved locations, which indicate estimation uncertainty in the monitored filed by making use of the observations at the sampled sites. As shown in the figures, the HGC method outperforms the other sampling planners by generating the lowest Kriging variance quantities in the study area.



Figure 4.4: Results of sampling locations, sampling paths, and mapping results using different survey planners.



40 50



10 20 30

40 50

10 20 30

Figure 4.5: Results of sampling locations, sampling paths, and results of estimation uncertainty using different survey planners.

10 20 30 40 50

4.4.3 Computational Cost

The computational efficiency is investigated as well in the simulation. The proposed method and the compared methods schedule their sampling missions depending on the determination of the SDR. The computational costs for all these planners are addressed from two main sources: 1) searching: searching of the OSDR such that the corresponding total path length is within the travel cost constraint; 2) planning: generation of the coverage path for sampling using the determined OSDR.

To achieve the densest sampling of coverage with its total travel cost not exceeding the power supply limit, the target SDR can be determined by the brute-force search. In the proposed method, the planner is driven by the hexagonal cell size that specifies the total number of SLoIs and further indicates the total path length. However, in view of the irregular contour of the study area, there is no analytical expression relating the cell size and the number of SLoIs. For illustrating the relationship between them, a brute-force search is conducted, and the result is shown in Figure 4.6. As shown, although they are not monotonous, the general trend is that the smaller cell size leads to a larger number of SLoIs as well as a longer path length. A binary search can be utilized to make this searching process approximate the OSDR. The search efficiency can be significantly improved but only a near-optimal SDR might be found since it may reach a local optimum as the total travel length is not strictly monotonous as a function of the cell size.



Figure 4.6: Relationships between cell size and number of SLoIs, and total length.

The processing times of these planners are shown in Figure 4.7, according to the power supply settings given in Table 4.1. The simulation was executed in Matlab R2016a on a desktop computer with an Intel(R) Core(TM) i7-6700K (4.00 GHz) processor and 32 GB of RAM. In the proposed hexagonal grid-based survey planners, the generation of a complete scheduling is not required when searching for the OSDR. As presented in Section 4.3.1, the search to find the optimal cell size depends on the resulting number of SLoIs as a function of a cell size using the ACC decomposition, which is solvable in polynomial time. Similarly, the grid-based methods including the SGSTC also have this advantage of speeding up the search process for the target SDR. In comparison, the HGTSP and the DMPP methods have to finish a complete planning process for each SDR to check if the resulting coverage path satisfies the constraint. The entire process considerably increases the computational complexity and the processing time for finding the OSDR using the brute-force search or even the binary search (see Figure 4.7 (a) and (b)).

After obtaining the OSDR, the proposed HGC survey planner generates the sampling path in polynomial time. It goes over all coarse cells and then covers the unvisited fine cells. In comparison, the SGSTC and the HGSTC approaches need to traverse all coarse cells to generate an MST first and revisit all fine cells to circumnavigate the MST. The DMPP approach generates the parallel transects in each decomposed cell in polynomial time and derives the proper sequence by connecting cells using a TSP solver, which is computationally intensive when the planning scale is high. In the simulation, the HGTSP approach was solved by linear integer programming (LIP). Figure 4.7 (c) and (d) show that the proposed HGC survey planner outperforms the other compared approaches in planning a sampling mission.

As clear from Figure 4.7 (e) and (f), the proposed planners show their computational efficiency over the existing methods. The experimental results verify that the proposed HGC method provides superior performance among the others in effective and efficient planning for unknown field estimation and environmental mapping, supporting its practical applicability in scheduling mobile sampling for environmental monitoring.



(a) Searching Time (brute-force search)



(b) Searching Time (binary search)

SGSTC

DMPP

HGTSP

10

9



(c) Planning Time (brute-force search)



(e) Total Processing Time (brute-force search)



60 SGSTC 55 DMPP 50 HGTSP Average Total Time (sec) HGSTC HGC 10 5 0 500 400 600 700 800 Energy Budget

(f) Total Processing Time (binary search)

Figure 4.7: Algorithm performance related to computational efficiency.

4.5 Field Test

The proposed planners and the planners with which they are compared have been evaluated by real-world field tests. They were implemented on the developed USV-based sensing platform (refer to Chapter 6.3) and evaluated at the Yosef Wosk Reflecting Pool of the University of British Columbia (refer to Chapter 3.4).

In the experiment, one mobile robot was implemented by applying one of the planners to carry out the corresponding sampling mission. Meanwhile, another robot was deployed to collect data in a denser distribution that was used as a validation dataset to evaluate the prediction performance. The fully charged 14.8 V, 10 Ah Lithium Polymer (LiPo) battery enabled a USV to run continuously for about 80 min (depending on the hydrodynamics of the monitored environment) at an average speed of 0.4 m/s. With this power condition, the total distance that a robot can travel is about 1920 m. The fully charged batteries were provided to both robots for each field test. 25% battery capacity was assigned to complete the sampling mission for the first robot, which indicates that the total travel distance is approximately 320 m. 40% battery capacity was assigned to the second mobile robot to collect validation data samples, which indicates that the total travel distance is approximately 768 m. The time cost t_M for the measurement process at each sampling location was set to 10 seconds. The updating time interval t_{udt} was set to 15 min.

Temperature and electrical conductivity are selected as the monitored parameters of interest. In the field tests, the underlying environmental models of the parameters were learned by observations from the first robot. The measurements taken by the second robot were utilized to validate the resulting prediction. Table 4.2 shows the performance of the RMSE by implementing the different survey planners. The monitored parameters, temperature and conductivity, are in units of Celsius and μ S/cm, respectively.

	SGSTC	DMPP	HGTSP	HGSTC	HGC
OSDR	2.68	2.39	4.46	4.18	4.18
Temperature RMSE	0.0517	0.0476	0.0334	0.0337	0.0331
Conductivity RMSE	0.1545	0.1429	0.1288	0.1282	0.1277

Table 4.2: RMSE performance of different survey planners in the field tests.

As shown in the table, all the RMSE results are relatively small. It is because the monitored pool has a limited spatial scale and displays unapparent spatial variation during experiments. The proposed approaches provide better prediction performance at the validation sites. Figure 4.8 provides the mapping results of the Yosef Wosk Reflecting Pool by making use of the observations from the generated HGC survey mission. The experimental results of the field tests further validate the conclusion in the numerical simulation that the proposed hexagonal grid-based survey planners outperform the state-of-the-art methods on field estimation and mapping.



Figure 4.8: Results of the proposed HGC survey planner.

(a) Generated coverage path for sampling; (b) Temperature mapping result; (c) The result of electrical conductivity mapping.

4.6 Summary

In the present chapter, the problem of energy-constrained survey planning was addressed. The presented planning strategies provided a reliable and efficient sampling path for mobile robot scheduling to survey an unknown environment of interest in a performanceoriented manner under a power constraint. The proposed planner guaranteed that the generated coverage path was length-bounded and energy-efficient, and could distribute coverage sampling densely and evenly across the overall surveillance area. The experimental results based on a real-world dataset validated the algorithm performance on mapping accuracy and computational efficiency. In practical applications, the proposed schemes can be implemented for spatial exploration, random field estimation, and environmental mapping.

Chapter 5: Informative Path Planning

This chapter addresses the problem of informative path planning for predicting an environmental field. The term "informative" is used in this context to mean "high content of useful information." Mobile sensing robots are scheduled to take data samples at the sites that are more informative among all possible sampling locations. A built-in GMRF is implemented to model the random field. A hierarchical planning framework is proposed, consisting of a global planner and a local planner. The global planner is operated at the sink to guide the robots to the most informative sites as well as the local planner is carried out on board each robot. The proposed framework provides a practical solution for a WMSN to achieve information-driven adaptive sampling for spatiotemporal field monitoring.

5.1 Overview

When implementing a robotic sensing system, owing to the limited number and mobility of the sensing robots, the sites that will be sampled within any time period is bounded. Furthermore, the resource constraints of the system (e.g., power supply) impede its operation life on both spatial and temporal scales. To capture the underlying environmental variation more effectively, mobile sensing robots are required to take samples at locations that are more representative and informative. With the prior knowledge that is gathered from an initial deployment, the underlying environmental model can be estimated with less uncertainty. In addition, redundant collections may not contribute to information gain but will use up the associated energy. Therefore, informative sampling design is important to sample and interpret the surveyed area more efficiently and effectively.

Information-theoretic sensing techniques have been derived by exploiting a statistical model of the monitored field. For instance, GP has been widely studied to interpret environmental phenomena and design optimal sensor placements [87] since it can make predictions over a spatial area by making use of finite observations, and also estimate the uncertainty at the predicted sites. Many information-theoretic criteria have been proposed for a GP model to measure the quality of an experimental design. Studies on information-theoretic quantities have indicated their power to exploit useful information

based on the GP model. For all these criteria, the optimization processes have been proven combinatorial NP-hard. The main disadvantage of the GP approaches is that factorizing the dense covariance functions and predicting the underlying field with growing number of measurements leads to high computational complexity [62].

Approaches of approximation have been studied to obtain a near-optimal solution while improving computational efficiency. For example, the near-optimal sensor placements in the GP model based on the MI metric was studied in the work of [62]. In addition, GMRF has been introduced to approximate the GP model [88]. Xu et al. [89] proposed a Bayesian spatial prediction method under the GMRF model to predict a large scale spatial field using mobile sensing agents. In their paper, adaptive sampling based on an entropy criterion was applied to improve the quality of prediction and reduce the estimation uncertainty of hyperparameters. The GMRF-based MI maximization problem was resolved to select the best spatial sensors in the work of [90]. Computation complexity of the MI-based GMRF approach is achieved mainly subject to the size of the precision matrix.

Many efforts in the field of optimal experimental design considered off-line optimization at a sink or a central server before the sensor deployment. They were mostly implemented in the WSNs with static sensors that were installed at fixed locations. In real-world applications, it is crucial for mobile sensing systems to provide the capabilities of online sampling, autonomous navigation, and real-time decision-making. The execution process of sampling design and path planning is required in real-time or at least near real-time. Although approximation model with a precision matrix of a GMRF model can facilitate computational efficiency, it is impractical to optimize the sampling planning problem using robotic sensors [91].

This chapter proposes a hierarchical informative path planner for a WMSN to gather information adaptively and efficiently for environmental field mapping. The proposed planner seeks the feasible scheme for bridging the gap between the intractable global informative optimization and the shortsighted local greedy planning. The GMRF model is built in the proposed scheme to represent the environmental scalar field and determine the MI-based most significant sites for adaptive sampling and spatiotemporal mapping.

5.2 Problem Statement

In the context of GMRF, the physical quantity of the random field is discretized and specified as:

$$\mathbf{Y}(\mathbf{s}) = \mathbf{X}(\mathbf{s}) + \mathbf{Z}(\mathbf{s}) + \boldsymbol{\varepsilon}(\mathbf{s}), \qquad (5.1)$$

where $\mathbf{X}(\mathbf{s})$ is a regression function that defines the mean value of the random process $\mathbf{Y}(\mathbf{s})$, $\mathbf{Z}(\mathbf{s})$ is a GMRF with a zero mean and a precision matrix \mathbf{Q} , i.e., $\mathbf{Z}(\mathbf{s}) \sim \mathcal{GMRF}(\mathbf{0}, \mathbf{Q})$, and $\boldsymbol{\varepsilon}(\mathbf{s})$ is an independent and identically distributed (i.i.d.) noise process denoted as $\boldsymbol{\varepsilon}(\mathbf{s}) \sim N(\mathbf{0}, \sigma_{\varepsilon}^{2} \mathbf{I}_{\varepsilon})$.

In the present work, a hierarchical informative path planner is proposed for a WMSN that consists of multiple mobile sensing robots and a sink (i.e., a base station). Let the set $U = \{u_1, u_2, ..., u_M\}$ represent $\mathcal{M} \in \mathbb{Z}^+$ robots that are indexed by $m = 1, 2, ..., \mathcal{M}$. Each robot follows its planned sampling path $p_m = (\mathbf{s}_{m,0}, \mathbf{s}_{m,1}, \mathbf{s}_{m,2}, ..., \mathbf{s}_{m,\mathcal{N}_m})$ with a sequence of sampling sites $\mathbf{s}_{m,k} \in S$, $k = 0, 1, 2, ..., \mathcal{N}_m$. Following a sampling path p_m , the robot takes a data sample at its current location $\mathbf{s}_{m,k}$ where a physical quantity is observed from a random variable that is expressed as:

$$Y(\mathbf{s}_{m,k}) = X(\mathbf{s}_{m,k}) + Z(\mathbf{s}_{m,k}) + \varepsilon(\mathbf{s}_{m,k}).$$
(5.2)

Let $y_{m,k}$ denote an observation of the random variable. Let S_t and \mathbf{y}_t denote the set of locations of the \mathcal{M} robots at time t and the corresponding measurements at time t, respectively. Let $S_{1:t}$ and $\mathbf{y}_{1:t}$ denote the locations of robots and the corresponding measurements from time 1 to time t, respectively. In the context of GMRF/SPDE method that is defined on a triangular mesh, the conditional mean (prediction) and the conditional precision matrix at the query locations $\tilde{S} \setminus S_{1:t}$ can be obtained as:

$$\hat{\mathbf{Y}}(\tilde{S} \setminus S_{1:t} \mid S_{1:t}) = \boldsymbol{\mu}(\tilde{S} \setminus S_{1:t} \mid S_{1:t}) = \boldsymbol{\mu}(\tilde{S} \setminus S_{1:t}) + \mathbf{Q}_{\tilde{S} \setminus S_{1:t} \mid S_{1:t}}^{-1} \mathbf{A}^{T} \mathbf{Q}_{\varepsilon} [\mathbf{y}_{1:t} - \mathbf{A}\boldsymbol{\mu}(S_{1:t})], \quad (5.3)$$

$$\mathbf{Q}_{\tilde{S}\backslash S_{1t}|S_{1t}} = \mathbf{Q}_{\tilde{S}\backslash S_{1t}} + \mathbf{A}^T \mathbf{Q}_{\varepsilon} \mathbf{A} .$$
(5.4)

where \mathbf{A} is the projector matrix that maps the mesh vertices to the selected locations. More details are referred to [57], [92].

To sample data adaptively, sites that are more informative should be visited. The MI, an information-theoretic quantity, is implemented as a metric to evaluate the "informativeness" of a sampling site. In the proposed planner, the target destinations to be visited are the most significantly informative sites conditioned on the statistical spatial field model, which are measured by the MI criterion (see Equation (2.12)). In the context of GMRF, the entropy and the conditional entropy in the MI can be obtained by the precision matrix [90], which are expressed as:

$$MI(Y(S), Y(\tilde{S} \setminus S)) = H(Y(\tilde{S} \setminus S)) - H(Y(\tilde{S} \setminus S) | Y(S))$$

= $-\frac{1}{2} \log (\det(\mathbf{Q})) + \frac{1}{2} \log (\det(\mathbf{Q}_{\tilde{S} \setminus S|S}))$ (5.5)

In a mobile robotic sensing process, each robot moves toward a target location that is planned based on the global information, and measures the sites that are locally more informative along the sampling trajectory over the travel duration. The target destination of the sampling path is generated at the sink by exploiting the global information over the study area. An anytime algorithm is an algorithm that can determine better and better solutions as the processing time increases. The optimal sampling sites are designed by an anytime algorithm at the sink over the entire GMRF spatial scale, which can guide the robots to collect data from the most informative sites. Concurrently, while heading for the optimal sites, the robots in the field measure from the informative locations along the path through local information greedy planning. In this manner, the hierarchical framework helps informative path planning and adaptive sampling in consideration of computational efficiency and prediction performance. The planning strategies are presented in the next section.

5.3 Hierarchical Informative Path Planner

To properly carry out the on-line sampling process, a mobile sensing agent is required to make fast decisions in planning. However, a robotic sensor in the field generally has limited on-board computational capability, which hinders the implementation of the MI-based optimization on it. A sink in a WMSN generally has a superior computational capability compared to the sensor nodes in the network. In the proposed hierarchical planner, the

target sites are generated over time by the sink and assigned to the robots as they move toward the goal location.

For a robot \mathcal{U}_m , the measurement process starts at its initial location $\mathbf{s}_{m,0}$ at time t = 0. For the following time steps, the robots heads to the first target optimal site that is designated as $\mathbf{s}_{m,1}^{(\text{opt})}$. Meanwhile, the local path is planned on-board at the robot by exploiting the local information greedily, which leads to a sequence of local sampling locations starting from $\mathbf{s}_{m,0}$ and ending at $\mathbf{s}_{m,1}^{(\text{opt})}$, i.e., $p_{m,1} = (\mathbf{s}_{m,0}, \mathbf{s}_{m,1}, \mathbf{s}_{m,2}, \dots, \mathbf{s}_{m,1}^{(\text{opt})})$. Subsequently, it heads to the next target site $\mathbf{s}_{m,2}^{(\text{opt})}$ from $\mathbf{s}_{m,1}^{(\text{opt})}$ with its corresponding sequence of local sampling locations $p_{m,2}$. Continuing on in this manner, the informative sampling path of the robot can be expressed as $p_m = (p_{m,1}, p_{m,2}, \dots, p_{m,j}, \dots)$, where $p_{m,2} = (\mathbf{s}_{m,k} = \mathbf{s}_{m,j-1}^{(\text{opt})}, \mathbf{s}_{m,k+1}, \dots, \mathbf{s}_{m,k+n} = \mathbf{s}_{m,j}^{(\text{opt})})$, $j \in \mathbb{Z}^+$, $\mathbf{s}_{m,0}^{(\text{opt})} = \mathbf{s}_{m,0}$. For notational simplicity, the subscript *m* is omitted in the remaining subsections.

5.3.1 Local Greedy Planner

The local greedy strategies to generate the sampling path from $\mathbf{s}_{j-1}^{(opt)}$ to $\mathbf{s}_{j}^{(opt)}$ are introduced now. The global planner to obtain the target site $\mathbf{s}_{j}^{(opt)}$ is presented in Section 5.3.2.

Given the starting location $\mathbf{s}_{j-1}^{(\text{opt})}$ and the next target location $\mathbf{s}_{j}^{(\text{opt})}$, $j \in \mathbb{Z}^{+}$, a local sampling path is planned on-board at the robot. The path is defined by a sequence as $p_{j} = (\mathbf{s}_{k}, \mathbf{s}_{k+1}, ..., \mathbf{s}_{k+n})$, where $\mathbf{s}_{k} = \mathbf{s}_{j-1}^{(\text{opt})}$, $\mathbf{s}_{k+n} = \mathbf{s}_{j}^{(\text{opt})}$, $n \in \mathbb{Z}^{+}$. Given a current location \mathbf{s}_{\cdot} , the next sampling location $\mathbf{s}_{\cdot+1}$ on the local sampling path is illustrated in Figure 5.1 (the dot in the subscript denotes any location index in the path sequence).



Figure 5.1: Generation of the next sampling location on the local path.

In the figure, the sampling locations are denoted by red stars, including the starting point $\mathbf{s}_k = \mathbf{s}_{j-1}^{(\text{opt})}$, the target location $\mathbf{s}_{k+n} = \mathbf{s}_j^{(\text{opt})}$, the current location \mathbf{s}_{\cdot} , and the next location $\mathbf{s}_{\cdot+1}$ that is planned to visit. To determine the location of $\mathbf{s}_{\cdot+1}$ given \mathbf{s}_{\cdot} , an arc \mathcal{R} is first constructed with the point \mathbf{s}_{\cdot} as the circle's center and a radius of r. The radius is determined by the product of the robot speed and the time interval of a data update. The constructed arc (the green arc in the figure) is symmetric with respect to the intersection point \mathbf{c}_{\cdot} (the black dot) between the arc and the line that connects \mathbf{s}_k and \mathbf{s}_{k+n} . The arc angle is set as θ . The arc is discretized to a set of points \widehat{S} . Then the next sampling location $\mathbf{s}_{\cdot+1}$ is generated by selecting the point with the maximum MI among the discretized points on the arc, which can be expressed as:

$$\mathbf{s}_{\bullet+1} = \arg\max_{\mathbf{s}\in\widehat{S}} MI(Y(\mathbf{s}), Y(S_{\widehat{A}} \setminus \mathbf{s})),$$
(5.6)

where \hat{A} defines the surrounding space that is involved by making this observation; $\tilde{S}_{\hat{A}}$ denotes the possible sampling locations within the area \hat{A} . The surrounding space \hat{A} for $\mathbf{s}_{,+1}$ is indicated in Figure 5.1 by the red dash line.

An execution example of the local planner is shown in Figure 5.2. In the figure, the blue stars indicate the discretized potential locations on the arcs. The red stars indicate the selected local optimal locations to be sampled. The blue line shows the local path that is planned from the starting point $\mathbf{s}_k = \mathbf{s}_{j-1}^{(\text{opt})}$ to the next target point $\mathbf{s}_{k+n} = \mathbf{s}_j^{(\text{opt})}$.



Figure 5.2: Execution example of local path planning.

Starting at the site $\mathbf{s}_{j-1}^{(\text{opt})}$ and heading to the next target site $\mathbf{s}_{j}^{(\text{opt})}$, the actions of a robot are listed below:

- Step 1: Start at the current location s, take a measurement and transmit it to the BS.
- **Step 2:** Check the next target site $\mathbf{s}_{j}^{(\text{opt})}$; If it is located within the radius r, i.e., $dis(\mathbf{s}_{.}, \mathbf{s}_{j}^{(\text{opt})}) \leq r$, set the site $\mathbf{s}_{j}^{(\text{opt})}$ as the next sampling location $\mathbf{s}_{.+1}$; otherwise, go to Step 3.
- **Step 3:** Create the arc that the robot can visit at the next time step, discretize the arc to points, calculate their corresponding MI values, set the point with the maximum value as the next sampling location $s_{.+1}$.

Step 4: Move to the next sampling location; repeat Step 1.

5.3.2 Global Planner of Near-Optimal Design

As introduced in the previous subsection, a robot operates the sampling mission by visiting the target sites according to the order $p_m = (\mathbf{s}_0^{(\text{opt})}, \mathbf{s}_1^{(\text{opt})}, ..., \mathbf{s}_{j-1}^{(\text{opt})}, \mathbf{s}_j^{(\text{opt})}, ...)$. For the optimal design of these target sites, a global planner is proposed by integrating the overall information at the sink. With the data collected by robots in the field, the sink updates the environmental model and determines the optimal sites to be measured, and then guides the sampling planning of the robotic sensors.

The initial environmental model can be estimated by the measurements of a prior survey. With prior knowledge of the environmental model, the optimal design in terms of the most informative sites over the entire space can be obtained. These sites are selected as the targets, which are added into the sampling missions of the robots in the field. In the GMRF model, the optimal sites can be derived by finding the locations across the entire space that have the maximum MI values. Although the GMRF approximation of finding the near-optimal solution can promote the algorithm efficiency, the computation is still intractable when designing a large number of optimal sites for mobile sensing even using the GMRF approach. Instead of obtaining a group of optimal sites at once, the proposed approach addresses the anytime property of the planning strategies at the sink.

A greedy algorithm is implemented to calculate the MI values of all the potential locations and find the optimal site conditioned on the historical sampled locations as well as the designed target sites. The generation of the next target site $\mathbf{s}_{j}^{(\text{opt})}$, starting at time $t = \mathcal{T}$, can be expressed as:

$$\mathbf{s}_{j}^{(\text{opt})} = \underset{\mathbf{s} \in \tilde{S} \setminus (S_{LT} \cup S_{Lj-1}^{(\text{opt})})}{\arg \max} MI(Y(\mathbf{s}), Y(\tilde{S} \setminus (S_{LT} \cup S_{Lj-1}^{(\text{opt})} \cup \mathbf{s}))), \qquad (5.7)$$

where $S_{1:\mathcal{T}}$ denotes the set of sampled locations from the beginning to time $t = \mathcal{T}$; and $S_{1:j-1}^{(\text{opt})}$ denotes the set of previous optimal designs, with $S_{1:j-1}^{(\text{opt})} \coloneqq \{\mathbf{s}_1^{(\text{opt})}, \mathbf{s}_2^{(\text{opt})}, \dots, \mathbf{s}_{j-1}^{(\text{opt})}\}$.

When executing to find the next target site $\mathbf{s}_{j}^{(\text{opt})}$ at the sink, the robots take more data samples concurrently. For example, while the sink starts to calculate the next target site $\mathbf{s}_{j}^{(\text{opt})}$ at time $t = \mathcal{T}$, the robots continuously make measurements at time $t = \mathcal{T} + 1, \mathcal{T} + 2,$ $\mathcal{T} + 3,...$ Consequently, the outcome of $\mathbf{s}_{j}^{(\text{opt})}$ that is derived at time $t = \mathcal{T} + n > \mathcal{T}$ by Equation (5.7) does not indicate the optimal design conditioned on the real sampled locations up to the current time, i.e., $S_{1:\mathcal{T}+n}$. To solve this problem, a sub-area removal strategy is proposed to consider the on-going measuring process when the sink is determining the next target site $\mathbf{s}_{j}^{(\text{opt})}$.

It is noted that the sampling locations that are scheduled by the local planner in the Section 5.3.1 are located around the line segment $\mathbf{s}_{j-1}^{(\text{opt})}\mathbf{s}_{j}^{(\text{opt})}$. The following proposition can be established.

Proposition 5.1: The maximum distance from a sampling location **s**. to the line segment $\mathbf{s}_{j-1}^{(\text{opt})}\mathbf{s}_{j}^{(\text{opt})}$ is $d_{\max} = 2 \cdot r \cdot \sin \frac{\theta}{2}$.

Proof: According to the design of the local planner in the Section 5.3.1, the first sampling location is placed at the site $\mathbf{s}_{j-1}^{(opt)}$, i.e., $\mathbf{s}_k = \mathbf{s}_{j-1}^{(opt)}$. The next sampling location $\mathbf{s}_{\cdot+1}$ can be derived iteratively, given the current location \mathbf{s}_{\cdot} . The auxiliary notations are shown in Figure 5.3, where ω is the angle between $\mathbf{s}_{j-1}^{(opt)}\mathbf{s}_j^{(opt)}$ and $\mathbf{s}_{\cdot}\mathbf{c}_{\cdot}$; and d_{\cdot} and $d_{\cdot+1}$ denote the distance from \mathbf{s}_{\cdot} and $\mathbf{s}_{\cdot+1}$ to the line segment $\mathbf{s}_{j-1}^{(opt)}\mathbf{s}_j^{(opt)}$, respectively. Considering $\omega \ge 0$, the next local sampling location is expressed as:

$$d_{\star+1} = \begin{cases} r \cdot \sin(\gamma - \omega) + d_{\star}, & \gamma \in [0, \theta] \\ r \cdot \sin(\omega - \gamma) - d_{\star}, & \gamma \in [-\theta, 0] \end{cases}$$

$$d_{\star} = r \cdot \sin \omega.$$
(5.8)



Figure 5.3: Next sampling location scheduled by the local planner.

For $\gamma \in [0, \theta]$, $d_{\star+1} = r \cdot [\sin(\gamma - \omega) + \sin \omega]$. It is obtained that when $\gamma = \theta$, $\omega = \frac{\theta}{2}$, the maximum distance is $d_{\max} | \gamma \in [0, \theta] = 2 \cdot r \cdot \sin \frac{\theta}{2}$. For $\gamma \in [-\theta, 0]$, $d_{\star+1} = 73$ $r \cdot [\sin(\omega - \gamma) - \sin \omega]$. It is obtained that when $\gamma = -\theta$, $\omega = 0$, the maximum distance is

$$d_{\max} \mid \gamma \in [-\theta, 0] = r \cdot \sin \theta$$
. Since $d_{\max} \mid \gamma \in [-\theta, 0] = r \cdot \sin \theta = 2 \cdot r \cdot \sin \frac{\theta}{2} \cos \frac{\theta}{2} \le d_{\max} \mid \gamma$

 $\in [0, \theta]$, the maximum distance is obtained by $d_{\max} = 2 \cdot r \cdot \sin \frac{\theta}{2}$. Figure 5.4 shows the scenario that $d_{\star+1} = d_{\max}$, where the red dash line in the figure indicates the farthest distance to the central line that a local sampling location can achieve.



Figure 5.4: Bounded area of possible sampling locations.

Considering $\omega \le 0$, the sampling location **s**. locates beneath the line segment $\mathbf{s}_{j-1}^{(\text{opt})}\mathbf{s}_{j}^{(\text{opt})}$. In this case, the results are symmetric to the above derivations with respect to the line segment. The maximum distance is the same, given by $d_{\text{max}} = 2 \cdot r \cdot \sin \frac{\theta}{2}$.

Then the following lemma can be stated.

Lemma 5.2: For a local sampling path from $\mathbf{s}_{j-1}^{(opt)}$ to $\mathbf{s}_{j}^{(opt)}$, the sampling locations that are scheduled by the local planner are located within a bounded sub-area.

Proof: The result comes directly from Proposition 5.1. Any sampling location \mathbf{s}_{\cdot} is located at one of the two sides of the line segment $\mathbf{s}_{j-1}^{(opt)}\mathbf{s}_{j}^{(opt)}$ within the area, where the maximum distance of the area contour to the line segment is $d_{\max} = 2 \cdot r \cdot \sin \frac{\theta}{2}$.

Due to the submodularity property of the sampling process [62], for a sampled location, the information in its surrounding area is reduced. Since the measuring process is operated along the sampling path, the possible information gain at the area near the local

sampling path is reduced. As a result, there is no need to resample this area again in the same sensing cycle. In the proposed global planner, the sub-area near the local sampling path is removed when generating the next target site. Assume that the radius *r* indicates the range that designates the reduced informative region if the circle center is sensed. The following proposition can be stated.

Proposition 5.3: The minimum reduced informative area that is covered by any two consecutive sampling locations is bounded, with a minimum distance $d_{\min} = r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{\max}) - r \cdot \sin \omega_{\max}$, from the area boundary to the line segment $\mathbf{s}_{j-1}^{(\text{opt})} \mathbf{s}_j^{(\text{opt})}$.

Proof: Figure 5.5 shows the scenario of any two consecutive sampling locations, i.e., **s**. and **s**_{.+1}. In the figure, γ denotes the angle between the line segments **s**_.**c**_. and **s**_{.+1}. The red and the blue circles denote the reduced informative area conditioned on both **s**_. and **s**_{.+1}, respectively; and **v**₁ and **v**₂ denote the two transect points of the two circles.



Figure 5.5: Two consecutive sampling locations.

Considering $\omega \ge 0$, the distances from the two transect points to the line segment $\mathbf{s}_{i-1}^{(\text{opt})}\mathbf{s}_i^{(\text{opt})}$ can be derived as follows.

$$d_{\mathbf{v}_{1}} = r \cdot \sin(\frac{\pi}{3} + \omega - \gamma) - r \cdot \sin \omega ,$$

$$d_{\mathbf{v}_{2}} = r \cdot \sin(\frac{\pi}{3} + \gamma - \omega) + r \cdot \sin \omega .$$
(5.9)

Then the minimal condition of the distance d_{v_1} can be obtained by simple calculus as follows (the derivation is given in the Appendix A.1):

$$\gamma = \arg\min_{\gamma} d_{\mathbf{v}_{1}} = \theta,$$

$$\omega = \arg\min_{\omega} d_{\mathbf{v}_{1}} = \omega_{\max} = \arcsin(2\sin\frac{\theta}{2}),$$

$$\min d_{\mathbf{v}_{1}} = r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{\max}) - r \cdot \sin\omega_{\max}.$$
(5.10)

The minimal condition of the distance d_{v_2} can be obtained by simple calculus as follows (the derivation is given in the Appendix A.1):

$$\gamma = \arg\min_{\gamma} d_{\mathbf{v}_{2}} = -\theta,$$

$$\omega = \arg\min_{\omega} d_{\mathbf{v}_{2}} = \begin{cases} \omega_{1} = 0 \text{ or} \\ \omega_{2} = \omega_{\max} \end{cases},$$
(5.11)
$$\min d_{\mathbf{v}_{2}} = \begin{cases} r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{1}) - r \cdot \sin \omega_{1} \text{ or} \\ r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{2}) - r \cdot \sin \omega_{2} \end{cases}.$$

Since $\min d_{\mathbf{v}_1} | \{ \gamma = \theta, \omega_{\max} \} \le \min d_{\mathbf{v}_2} | \{ \gamma = -\theta, \omega_1 = 0 \}$ and $\min d_{\mathbf{v}_1} | \{ \gamma = \theta, \omega_{\max} \}$ $\le \min d_{\mathbf{v}_2} | \{ \gamma = -\theta, \omega_2 = \omega_{\max} \}$ (the derivation is given in the Appendix A.1), the minimum distance is obtained by $d_{\min} = \min d_{\mathbf{v}_1}$.

Considering $\omega \leq 0$, the sampling location **s**. locates beneath the line segment $\mathbf{s}_{j-1}^{(\text{opt})}\mathbf{s}_{j}^{(\text{opt})}$. In this case, the results are symmetric to the above derivations with respect to the line segment. In summary, by sampling at the two consecutive locations, two circles indicate the reduced informative area that they covered, which has a minimum occupied area. The minimum distance from the area boundary to the line segment $\mathbf{s}_{j-1}^{(\text{opt})}\mathbf{s}_{j}^{(\text{opt})}$ is d_{\min}

$$= r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{\max}) - r \cdot \sin \omega_{\max}, \text{ where } \omega_{\max} = \arcsin(2 \cdot r \cdot \sin \frac{\theta}{2}). \blacksquare$$

There is a minimum reduced informative area that is occupied by any two consecutive sampling locations. Accordingly, there is a region between these two locations, which is less informative after these points are sampled, no matter where these two points are located away from the line segment $\mathbf{s}_{j-1}^{(opt)}\mathbf{s}_{j}^{(opt)}$.

Theorem 5.4: Any local sampling path from $\mathbf{s}_{j-1}^{(opt)}$ to $\mathbf{s}_{j}^{(opt)}$ covers a dumbbell-shape area $A_{j-1,j}^{(opt)}$ where the corresponding region has less information, if the sampling mission along the local path is completed.

Proof: The result comes directly from Proposition 5.3. The minimum reduced informative area that is covered by any local sampling path is bounded in a dumbbell shape with three main regions, as shown in Figure 5.6. Two circles respond to the sampled locations $\mathbf{s}_{j-1}^{(\text{opt})}$ and $\mathbf{s}_{j}^{(\text{opt})}$. The middle rectangular region responds to the sampled locations along the local path, which has the minimum distance $d_{\min} = r \cdot \sin(\frac{\pi}{3} - \theta + \omega_{\max})$ $-r \cdot \sin \omega_{\max}$ from the boundary to the line segment $\mathbf{s}_{j-1}^{(\text{opt})} \mathbf{s}_{j}^{(\text{opt})}$.



Figure 5.6: Dumbbell-shaped area.

When generating the next target site $\mathbf{s}_{j}^{(\text{opt})}$, the reduced informative region $A_{j-1,j}^{(\text{opt})}$ defined by the previous sampling paths should be removed from the entire space of interest, which is defined as the Sub-Area Removal (SAR) strategy. By implementing SAR, the generation of the next target site $\mathbf{s}_{j}^{(\text{opt})}$, starting at time $t = \mathcal{T}$, can be expressed as:

$$\mathbf{s}_{j}^{(\text{opt})} = \arg\max_{\mathbf{s}\in\tilde{S}\backslash\tilde{S}} MI(Y(\mathbf{s}), Y(\tilde{S}\setminus(\tilde{S}\cup\mathbf{s}))), \qquad (5.12)$$

where $\tilde{S} \subseteq A_{1:j-1}^{(opt)}$ denotes the potential sampling locations within the removal area $A_{1:j-1}^{(opt)}$. After obtaining the target site $\mathbf{s}_{j}^{(opt)}$, it is assigned to the robot u^{*} for which the last target site of its sampling mission has the shortest distance to the generated site $\mathbf{s}_{j}^{(opt)}$.

The actions of the global planner at the sink are listed as follows:

- **Step 1:** Set the current timer t = T, Get the historical sampling mission with the sampled locations $S_{1:T}$ and the designed sites $S_{1:j-1}^{(opt)}$.
- **Step 2:** Determine the next target site $\mathbf{s}_{j}^{(opt)}$ with maximum MI value conditioned on the historical sampling mission.
- **Step 3:** Add the target site $\mathbf{s}_{j}^{(\text{opt})}$ into the sampling mission of the robot u^{*} , update the sampling mission.
- Step 4: Send the new sampling mission to the robot, repeat Step 1.

By following the proposed hierarchical scheme, the global planner in the sink obtains a near optimal design while the robots in the field carry out local planning and measuring simultaneously. Notice that the global planner is executed to make sure that the target sites can be implemented to navigate the robots in the field. In addition, with the continuous running of the global planning, the area of interest that is left to determine the optimal sites becomes smaller, which leads to faster execution of the optimal design over time.

The environmental phenomena may vary in two ways: 1) the concentrations of the parameters may change over time, but the model of the spatial correlation remains; 2) the physical quantities of the parameters change and the spatial correlations change as well. The former case indicates that the environmental model remains the same, which refers to the same optimal sampling design in each sensing cycle. In this case, the robots should complete a sensing cycle within a time interval t_{utd} while following the previously designed sampling paths after the previous cycle. The latter case indicates that the environmental model updating process is required. With on-line measurements, the environmental model can be changed to interpret the variance of the

spatiotemporal correlation over the random field. By using the historical measurements at the sampled sites, the hyperparameters can be re-estimated on-line to update the environmental model and re-design the optimal sampling sites. The model updating and adaptive sampling design are operated on-line and iteratively to interpret the spatiotemporal variance of the environment.

The flow chart of the overall hierarchical planner is shown in the Figure 5.7. Two treads are operated at the sink: anytime planning and model updating. In anytime planning, the global planner continuously generates the next target site and updates the new sampling mission until it is interrupted by the timer. The timer is reset at the initial time, or when the environmental model is updated. The remote server gathers the collections from the sink and reconstructs the environmental map at each time step.



Figure 5.7: Flow chart of the hierarchical informative path planner.

5.4 Numerical Experiments and Discussion

In this section, the proposed hierarchical planner is studied through a numerical experiment to evaluate its computational efficiency, prediction accuracy, and algorithm reliability.

The numerically generated spatial field is used in the simulation. The physical quantity of the spatial field is designated on a grid with 80 × 80. The mean function is set at X = 20. The hyperparameters of the precision matrix are chosen to be $\theta = (\tau, \kappa) = (1, 0.1)$. The noise level is set at $\sigma_{\varepsilon}^2 = 0.2$. In real-world applications, the mean function, parameters, hyperparameters, and the noise condition can be learnt by implementing the prior survey, as presented in Chapter 4. The generated spatial field is shown in Figure 5.8, which is used as the ground truth in the simulation.



Figure 5.8: Numerically generated spatial field.

The spatial field is discretized by a triangle mesh using the INLA package in R [92]. An example of the constructed mesh is shown in Figure 5.9. In the simulation, to design the most informative target sites in the global planner, the spatial field is discretized on the triangular mesh by approximately 350 vertices of triangles. In the local planner, the most informative sampling location is determined in each local planning out of 10 discretized locations on the arc. In the simulation, three mobile sensor agents ($\mathcal{M}=3$) are assigned to

make observations, starting from the initial locations that are randomly selected as $\mathbf{s}_{1,0} = (39,11)$, $\mathbf{s}_{2,0} = (45,75)$, $\mathbf{s}_{3,0} = (6,70)$). The travel distance of the agent between two consecutive samples is set at r = 3. The robot speed is set at v = 0.4 m/s. The measurement at a sampling location is set at $t_M = 10$ s. The simulation is executed in R on a desktop computer with an Intel(R) Core(TM) i7-6700K (4.00 GHz) processor and 32 GB of RAM.



Figure 5.9: Triangle mesh of the discretized spatial field.

The developed hierarchical path planner that incorporates the proposed sub-area removal strategy (HPP-SAR) is evaluated on the numerically generated scalar field. It provides an on-line solution for combining global sampling selection optimization and local greedy search for adaptive sampling and field mapping. In the experiment, a distributed path planner using a local greedy search strategy (DPP-LGS) and a central sampling planner through global optimization (CSP-GO) are compared to the HPP-SAR approach concerning the algorithm performance of the field prediction and the computational time cost.

In the DPP-LGS algorithm, each robot generates the next sampling site by finding the location that has the maximum MI or entropy quantity among the other sites around the current location. This strategy has been studied in past work (e.g., [15], [51], [89]) to achieve on-line planning for adaptive sampling and field mapping. Specifically, a circular arc (with the current robot location as the circle center) is discretized to multiple points that are designated as possible locations to be sampled, i.e., sampling candidates, for the next step. The circle radius is determined by the mobility of the robot. The mesh points in a local area around the current location are considered to evaluate the MI gains of the sampling candidates. In the experiment, for the DPP-LGS algorithm, the circle radius is set to r = 3; the number of discretized points is set to 22; and the local area is defined as a circular area with a radius of $2 \cdot r$.

Data is collected at the generated sampling locations and then utilized to predict the scalar field by utilizing the regression model of GMRF. RMSE is implemented as the measure to evaluate the prediction results, which is defined as:

$$RMSE(t) = \sqrt{\frac{1}{|I|} \sum_{\mathbf{i} \in I} \left[\hat{Y}(\mathbf{i}) | \mathbf{y}_{1:t} - y(\mathbf{i}) \right]^2} .$$
(5.8)

where *I* denotes the set of the grid nodes for spatial interpolation; $y(\mathbf{i})$ denotes the ground truth value at a location $\mathbf{i} \in I$.

Figure 5.10 shows the prediction performance by considering the variation of RMSE with time. As can be seen, the RMSE reduces gradually with the increased observations over time.



Figure 5.10: RMSE performance of resulting predictions.

Prediction outcomes are updated with time. The resulting predictions of the spatial field by implementing the HPP-SAR and the DPP-LGS algorithms at time t = 300 s, 600 82

s, and 900 s are shown in Figure 5.11 and Figure 5.12, respectively. In the figures, the randomly selected initial locations, planned target sites, and the sampled locations along the path are shown in hollow circles, valid circles, and star markers, respectively. The colors illustrate the physical quantities over the scalar field. As shown, the resulting predictions become closer to the ground truth as the time increases.



Figure 5.11: Mapping results of the proposed planner at different time (second).

As shown in Figures 5.10, 5.11, and 5.12, the proposed HPP-SAR planner can provide superior mapping results in a spatial field when compared with the DPP-LGS. In particular, the DPP-LGS approach can provide better performance at the initial stage of the sampling process (see Figure 5.10, t = 100 s, 200 s). It is because the distributed local planner is able to rapidly issue the sampling sites and start to guide the robots for data 83 collection. In comparison, the proposed HPP-SAR approach has to calculate the first target site before the robots start to take data samples. However, as time goes, the proposed algorithm achieves better prediction results since it can guide the robots to visit the globally optimized target sites by considering the historical measurements. The local greedy policy of the DPP-LGS method may remain trapped in a small region for some period of the sampling mission.



Figure 5.12: Mapping results of the DPP-LGS approach at different time (second).

The CSP-GO optimizes the design of sampling locations globally by the GMRF model, which ranks all the possible sites based on their MI quantities and then selects the best ones as the sites of interest. This strategy has been studied in past work [90]. However,

even without considering the path planning issue, the experimental design process is computationally intensive and practically infeasible due to the intractable computation of an approximated optimal solution. The number of selected sites, the corresponding processing time, and the corresponding predication performance are shown in Figure 5.13.



Figure 5.13: Algorithm performance of the CSP-GO approach.

(a) Computational time with respect to the number of selected sites. (b) RMSE with respect to the number of selected sites. (c) An execution example of the mapping result of 10 optimized sampling sites. (d) An execution example of the mapping result of 30 optimized sampling sites.

Remark 5.6: The proposed algorithm generates the global near-optimal sites in an anytime planning manner. For the computational procedure of each target site, the processing time decreases as the number of measurements increases.

The remark can be examined through the experimental results. As expected, the planner determines the most informative site one by one, and the size of the potential sites on the mesh decreases over time. Figure 5.14 illustrates, at time t = 900s, the order in which the target sites were generated and their assignment conditions for the three agents. In the figure, the star markers and the line segments indicate the sampled locations and the traveled path, respectively. The black dots indicate the generated target sites, which are labeled in the order of their generation time. Table 5.1 summarizes the processing time for generating the target sites. Notice that the sites have obtained persistently, which could also be interrupted at any time, providing the associated solutions up to that time. Also, they were derived increasingly faster from 48.89s for the first site to the 34.13s for the last one. The experimental results support the concluding statement in Remark 5.6.



Figure 5.14: Generated target sites and sampling paths at time t = 900s.

Target	Starting	Ending	Duration	Assigned	Previous	Visited
Site	Time	Time		Robot	Site	Time
	(second)	(second)	(second)			(second)
$\mathbf{S}_{1}^{(\mathrm{opt})}$	0	48.89	48.89	<i>u</i> ₂	$\mathbf{S}_{2,0}$	243.78
$\mathbf{S}_{2}^{(\text{opt})}$	48.89	96.65	47.76	u_1	$\mathbf{S}_{1,0}$	212.70
$\mathbf{S}_{3}^{(\mathrm{opt})}$	96.65	143.63	46.98	<i>u</i> ₃	s _{3,0}	318.81
$\mathbf{s}_4^{(\text{opt})}$	148.63	192.74	44.11	<i>u</i> ₂	$\mathbf{s}_{1}^{(\mathrm{opt})}$	379.54
$\mathbf{s}_5^{(\text{opt})}$	192.74	236.38	43.64	<i>u</i> ₁	$\mathbf{s}_{2}^{(\mathrm{opt})}$	387.88
$\mathbf{s}_{6}^{(\text{opt})}$	236.38	279.24	42.86	<i>u</i> ₂	$\mathbf{s}_{4}^{(\mathrm{opt})}$	436.46
$\mathbf{s}_7^{(\text{opt})}$	279.24	321.41	42.17	<i>u</i> ₃	$\mathbf{s}_{3}^{(\mathrm{opt})}$	378.33
$\mathbf{s}_8^{(\mathrm{opt})}$	321.41	363.35	41.94	u_1	$\mathbf{s}_{5}^{(\mathrm{opt})}$	444.80
$\mathbf{s}_{9}^{(\mathrm{opt})}$	363.35	404.44	41.09	<i>u</i> ₃	$\mathbf{s}_7^{(\mathrm{opt})}$	454.96
$\mathbf{s}_{10}^{(\mathrm{opt})}$	404.44	444.58	40.14	<i>u</i> ₁	$\mathbf{s}_8^{(\mathrm{opt})}$	639.69
$\mathbf{s}_{11}^{(\mathrm{opt})}$	444.58	484.18	39.60	<i>u</i> ₂	$\mathbf{s}_{6}^{(\mathrm{opt})}$	552.51
$\mathbf{s}_{12}^{(\text{opt})}$	484.18	523.39	39.21	<i>u</i> ₂	$\mathbf{s}_{11}^{(\mathrm{opt})}$	826.24
$\mathbf{S}_{13}^{(\mathrm{opt})}$	523.39	561.20	37.81	<i>u</i> ₃	$\mathbf{s}_{9}^{(\mathrm{opt})}$	896.08
$\mathbf{S}_{14}^{(\mathrm{opt})}$	561.20	596.61	35.41	u_1	$\mathbf{s}_{10}^{(\mathrm{opt})}$	N/A
$\mathbf{s}_{15}^{(\mathrm{opt})}$	596.61	630.74	34.13	<i>u</i> ₂	$\mathbf{s}_{12}^{(\mathrm{opt})}$	N/A

Table 5.1: Target sites and their corresponding assignments.

N/A: The target sites have not been visited within the time t = 900s.

5.5 Summary

This chapter proposed a new framework to plan informative paths, which was implemented in a WMSN to carry out adaptive sampling for spatiotemporal field mapping. The proposed framework provided a hierarchical scheme that consisted of a local greedy planner and a global near-optimal planner. The local planner operated on-board planning to plot the more informative locations along the local sampling path. Concurrently, the global planner at the sink, with its beneficial characteristics as an anytime algorithm, determined the subsequent destinations that were the most informative sites conditioned on the sampled locations and the previous targets. A new sub-area removal strategy was introduced in the present chapter to solve the growing set of the sampled locations during the global optimization at the sink.

Chapter 6: On-line Monitoring of Surface Water

This chapter presents the design and development of a rapidly deployable USV-based sensing platform for on-line monitoring of surface water. First, an on-line water quality index is developed, which is applicable in the quality indexing of data that are collected through automated sampling. Next, the details of the platform components are presented. The proposed survey planner and the on-line water quality index are integrated into the developed platform. To validate its performance, the platform is deployed in a real-world environment of water quality monitoring. The experimental results and the system performance are presented and discussed.

6.1 Overview

Recent advances in sensing, processing, and communication, and their integration into practical devices have resulted in significant progress in remote environmental monitoring through automated sampling and continuous on-line data collection. In-situ monitoring platforms with multiple sensors have been designed and deployed [1], [7] for field data collection at relatively high speeds (in the scale of minute or hour). The implementation of these devices has made possible the acquisition of adequate, accurate, and current data of water quality parameters for fast evaluation of water quality. However, the on-line data as generated through automated sampling present new requirements and challenges in applying the quality index method for data aggregation.

Practically all the existing WQIs, including the CCME, have been applied for longterm quality evaluation using data from human-intensive filed measurements at low sampling rates (in the scale of month to quarter). In this backdrop, a practical, realistic, reliable, and effective WQI is needed to handle on-line data, primarily for short-term water quality evaluation. An Online Water Quality Index (OLWQI) is developed in the present chapter to represent a large amount of online data for water quality indexing. It is modified by the index formulation of the state-of-the-art index, the CCME WQI, which is a widely used index and developed by the Canadian Council of Ministers of Environment (CCME) [71] for off-line indexing.
This chapter also presents the development of a rapidly deployable and easily maintainable WMSN platform for surface water monitoring. The rapid deployment framework has to be fast and easy to deploy and maintain [93]. It may be deployed in the field only for relatively short time periods (in the scale of a day to a week), but can achieve high-resolution spatiotemporal sampling. The objective of the developed platform is to provide effective and efficient quality evaluation of surface water at a high resolution in spatiotemporal manner. Compared to the state-of-the-art systems, the developed platform has superior performance on several aspects. First, the low cost of the components in the platform provide a cost effective solution for automated water quality evaluation. In addition, the developed platform is characterized by fast deployment and easy maintenance, which simplifies the initial deployment and follow-up maintenance procedures. Second, multiple MSNs facilitate area surveillance on the spatiotemporal scale unlike the static sensing stations. More importantly, the implemented planning algorithm provides an efficient path planner by incorporating energy and time constraints. Third, online quality indexing is implemented in the platform by integrating online data of multiple parameters to give a comprehensive, quite representative, and unbiased quality evaluation of a study area.

6.2 On-line Water Quality Index

A WQI provides an intuitively clear and adequately representative way to interpret the water quality by aggregating all the measured data of water quality parameters into a numerical score. Then, the score is classed into a clear quality category for reporting to the technicians, managers, policy-makers, general public, and other users. The CCME WQI has been widely used in water quality monitoring programs by many agencies and institutes throughout the world [72], [73], [75], [94]. It is generally applied off-line, by using data that are collected at low sampling rates (in the scale of month to quarter). Although CCME WQI has been used as a possible indexing approach for data collected through automated sampling [95], according to our experiments, disadvantages exist when directly implementing it for online quality indexing.

The OLWQI, which is developed in the present work, provides effective indexing results with a reliable sensitivity factor for large quantities of online data collected through

automated sampling. The index formulations of the OLWQI are expressed in analytical form to facilitate the automatic execution on implemented platforms.

The proposed OLWQI has been implemented on a real mobile sensing platform, which consists of a group of mobile sensor nodes, a base station located on shore, and a remote server. Spatiotemporal measurements and the on-line quality index are provided as the monitoring results, which will be utilized for further decision-making, policy-making, and water management.

The CCME WQI focuses on off-line evaluations, and utilizes data collected at a low sampling rate (typically at monthly or quarterly intervals) and at a limited number of sampling locations. The index formulation of the CCME WQI incorporates three statistical factors by comparing the measurements of water quality parameters and their guidelines (a range of acceptable values). The index formulation is based on the following three assessment factors [71]:

• **Scope** assesses the percentage of water quality parameters that do not meet their guidelines over the time period of interest:

$$F_1 \text{ (Scope)} = \frac{Number of failed parameters}{Total number of parameters} \cdot 100, \tag{6.1}$$

• **Frequency** represents the percentage of individual tests that failed the acceptable ranges over the time period of interest:

$$F_2 \text{ (Frequency)} = \frac{Number of failed tests}{Total number of tests} \cdot 100, \tag{6.2}$$

- Amplitude measures the degree by which the failed tests deviated from the acceptable levels. It is calculated in three steps as follows:
 - The fractional deviation of a failed test value from its acceptable limit is termed an *"Excursion"*. When the failed test value is greater than the acceptable upper limit of its relative guideline:

$$Excursion_{i} = \frac{Failed \ test \ value_{i}}{Guideline_{i}} - 1, \tag{6.3}$$

When the failed test value is less than the allowable lower limit of its relative guideline:

$$Excursion_{i} = \frac{Guideline_{i}}{Failed \ test \ value_{i}} - 1, \tag{6.4}$$

 The Normalized Sum of Excursions (NSE) is defined as the average value of excursions:

$$NSE = \frac{\sum_{i=1}^{n} Excursion_{i}}{Total \ number \ of \ tests},$$
(6.5)

3) Amplitude Factor scales the *NSE* to a value in the range 0-100 by an asymptotic function:

$$F_3$$
 (Amplitude) = $\frac{NSE}{NSE+1} \cdot 100$, (6.6)

• The final index, ranging from 0 to 100 with a higher score representing a better quality, is calculated as the root mean square of these three factors as follows:

$$CCME WQI = 100 - \sqrt{\frac{F_1^2 + F_2^2 + F_3^2}{3}}, \qquad (6.7)$$

In this section, the original index formulation of the CCME WQI is first expressed in an analytical form. Using that, it may be applied (with possible modification) to automated online data measurement at a relative high sampling rate (at minute or hour intervals). Furthermore, the derivation of the analytical form facilitates its further implementation on the wireless mobile sensor network for online water quality indexing.

In the data acquisition process of the present work, the data records are measured and logged one by one. Each data record contains the sampling locations, sampling time, and the measurements of multiple water quality parameters. Each measurement is called a data sample or a test. The measurements involved in the indexing are expressed as:

_

$$\mathbf{y} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1\ell} \\ y_{21} & y_{22} & \cdots & y_{2\ell} \\ \vdots & \vdots & \vdots & \vdots \\ y_{f1} & y_{f2} & \cdots & y_{f\ell} \end{bmatrix}_{f \times \ell},$$
(6.8)

where a row vector represents the measurements of ℓ parameters of a data record, and f presents the number of samples involved in the index calculation. A guideline matrix is defined as:

$$\mathbf{G} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1\ell} \\ g_{21} & g_{22} & \cdots & g_{2\ell} \end{bmatrix}_{2\times\ell},$$
(6.9)

where $g_{1,\ell}$ and $g_{2,\ell}$, respectively represent the lower and upper acceptable values of the ℓ parameters, according to their guidelines. For example, pH value, a common water quality parameter, is specified as $g_{1,pH} = 6.5$ and $g_{2,pH} = 9$ for the protection of aquatic life in freshwater, according to the Canadian Environmental Quality Guidelines (CEQG) [96].

To describe whether a data sample in **y** has failed compared to its guideline, a Boolean variable b_{ij} corresponding to a Boolean function $B(m_{ij})$, i = 1, ..., f, $j = 1, ..., \ell$, is defined as:

$$b_{ij} = B(m_{ij}) = \begin{cases} 0 & m_{ij} \in [g_{1j}, g_{2j}] \\ 1 & m_{ij} \notin [g_{1j}, g_{2j}] \end{cases}.$$
(6.10)

According to this formulation, the number of failed parameters is given by $\mathcal{P} = \sum_{j=1}^{\ell} \min_{1 \le i \le \ell} (b_{ij})$; and the number of failed tests is given by $\mathcal{Q} = \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} b_{ij}$. Then, the three assessment factors in the CCME WQI can be analytically expressed as:

$$F_{1} (\text{Scope}) = \frac{\mathcal{P}}{\ell} \cdot 100,$$

$$F_{2} (\text{Frequency}) = \frac{\mathcal{Q}}{f \cdot \ell} \cdot 100$$

$$Excursion_{ij} = \begin{cases} \left(\frac{g_{1j}}{y_{ij}}\right) - 1 & y_{ij} < g_{1j} \\ 0 & g_{1j} \leq y_{ij} \leq g_{2j} \\ \left(\frac{y_{ij}}{g_{2j}}\right) - 1 & y_{ij} > g_{2j} \end{cases}$$

$$(6.11)$$

$$(\text{Amplitude}) = \frac{\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} Excursion_{ij}}{\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} Excursion_{ij} + f \cdot \ell} \cdot 100$$

The impact of a failed test, $m_{ij} \notin [g_{1j}, g_{2j}]$ on the three factors will be different since a failed test causes different score changes in the three factors. In this work, the increment ΔF in a factor score caused by a failed test is defined as the factor sensitivity S. The sensitivities of the three assessment factors are given in Table 6.1.

Table 6.1: Factor sensitivity of the CCME WQI.

Factor	Factor Sensitivity					
Scope	$S_1 = \frac{1}{\ell} \cdot 100$					
Frequency	$\mathcal{S}_2 = \frac{1}{\int \cdot \ell} \cdot 100$					
Amplitude ¹	$S_{3} = \frac{f \cdot \ell}{\sum Excursion + f \cdot \ell + \frac{\left(\sum Excursion + f \cdot \ell\right)^{2}}{Excursion_{ij}}} \cdot 100$					

¹ The derivation of the sensitivity is given in the appendix.

 F_3

According to the sensitivity expressions in Table 6.1, biased factor sensitivity may exist when it is applied to handle large amounts of online measurements ($f \gg l$). On one hand, the Scope Factor may dominate the index score with only a few failed tests. For example, a failed test among the total of $f \cdot l$ tests produces the change $\Delta F_1 = \frac{1}{l} \cdot 100$ in the first factor, which is f times larger than the change in the second factor $\Delta F_2 = \frac{1}{f \cdot l} \cdot 100$, for the same cause. On the other hand, the parameter with a wide data range may lead to a score bias in the third factor, according to the expressions of the terms *Excursion*_{ij} and S_3 . For example, failed electrical conductivity tests (data ranging from 0 to 14) in a large number of data records.

To avoid the score bias in the Scope Factor, the average of all measurements of a parameter, $\overline{y}_{,j}$, is used for comparison with its guideline. Specifically, the Scope Factor is modified as:

$$F_1'(\text{Scope}) = \frac{\sum_{j=1}^{\ell} B(\overline{y}_{,j})}{\ell} \times 100, \qquad (6.12)$$

To avoid the score bias in the Scope Factor, the average of all measurements of a parameter, $\overline{y}_{,j}$, is used for comparison with its guideline. Specifically, the Scope Factor is modified as:

$$Excursion'_{ij} = \begin{cases} \frac{g_{1j} - y_{ij}}{g_{1j} - MIN_{j}} & y_{ij} < g_{1j} \\ 0 & g_{1j} \le y_{ij} \le g_{2j} \\ \frac{y_{ij} - g_{2j}}{MAX_{j} - g_{2j}} & y_{ij} > g_{2j} \end{cases}$$
(6.13)

where *MAX* and *MIN* represent the maximum and the minimum values, respectively, that a certain parameter can reach. In the original formulation, $Excursion_{ij} \in [0, \frac{MAX_j}{g_{2j}} - 1]$ or $[0, \frac{g_{1j}}{MIN_j} - 1]$. In the modified formulation, *Excursion'*_{ij} $\in [0, 1]$ for all parameters. Thus,

parameters with different variation ranges will result in an unbiased index score within the modified Amplitude Factor. The modified Amplitude Factor is derived using the modified term *Excursion'*_{ij} given by:

$$F'_{3} \text{ (Amplitude)} = \frac{\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} Excursion'_{ij}}{\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} Excursion'_{ij} + \ell} \times 100, \qquad (6.14)$$

Note that the $f \cdot \ell$ term in the original formulation (6.11) is replaced by ℓ in the modified formulation. The aim is to reduce the high weighting of this term, as introduced by a large volume of online measurements. The Frequency Factor remains in its original form, as it represents the frequency of failed tests with a reasonable factor of sensitivity. The sensitivities of the modified factors are given in Table 6.2. This modified index is the Online Water Quality Index (OLWQI):

$$OLWQI = 100 - \sqrt{\frac{F_1'^2 + F_2'^2 + F_3'^2}{3}},$$
(6.15)

Factor	Factor Sensitivity					
Scope	$\mathcal{S}_{1}' = \frac{1}{f \times \ell} \times 100$					
Frequency	$\mathcal{S}_2' = \frac{1}{f \times \ell} \times 100$					
Amplitude	$S_{3}' = \frac{\ell}{\sum Excursion' + \ell + \frac{\left(\sum Excursion' + \ell\right)^{2}}{Excursion'_{ij}}} \times 100$ Excursion'_{ij} \in [0,1]					

Table 6.2: Factor sensitivity of the Online Water Quality Index (OLWQI).

6.3 Development of a WMSN for On-line Monitoring of Surface Water

In this section, the developed WMSN is presented in detail. The system consists of multiple USVs that are incorporated as the Mobile Sensor Nodes (MSN) in the network, a Base Station (BS) on the bank that works as a sink of the local wireless network, and a Remote Server (RS).

In the implementation, the USVs are deployed in a distributed way in the monitored field. The survey missions (sampling locations and moving paths) for the USVs are generated at the RS and transmitted to the USVs via the BS. Then, the USVs follow the received missions to collect data at the scheduled sampling locations. Each USV consists of a set of heterogeneous sensors to measure different water quality parameters. The collected data is then transmitted to the BS through a local wireless network.

The monitoring results are presented at the BS via a Local Assessment Unit (LAU) in two forms: (1) the field map in terms of the water quality parameters at the sampling locations; and (2) the online water quality index. The former form presents the quantitative measurements in the field. The latter form presents the qualitative evaluation of the surface water. The results are also transmitted to the RS which contains a Central Assessment Unit (CAU). Thus, the monitoring results can be accessed locally at the BS by the technicians in the field or accessed remotely by the users via the Internet. The architecture of the proposed WMSN is illustrated in Figure 6.1.



Figure 6.1: The proposed wireless mobile sensor network.

6.3.1 Mobile Sensor Nodes

Various water parameters can be monitored through automated sensing [97]. These parameters include flow rate, temperature, air pressure, pH value, dissolved oxygen, electrical conductivity, oxidation-reduction potential, nitrogen, phosphate, organic matter, microorganisms, and so on. Selection of the water quality parameters is based on the specific end use and the monitoring objective. In the developed platform, five sensors are implemented in each MSN to measure five representative parameters. They are listed below.

- **Temperature sensor** (**T**) senses water temperature through a thermoresistive probe whose resistance increases with the heat transferred from the aquatic source. Many parameters are affected by temperature. Thus, temperature compensation is required during the sensor calibration for those parameters.
- **pH Value sensor** (**pH**) measures the output voltage of an electrode due to the hydrogen ion activity in the water, which can then be translated into the pH value according to the hydrogen ion concentration.
- **Dissolved Oxygen (DO)** sensor measures the output voltage of the sensor with an anode and a cathode, which is proportional to the concentration of the dissolved oxygen in the water.
- Electrical Conductivity (EC) sensor measures the resistance of a two-pole cell of the sensor. Water conductivity is proportional to the conductance (the inverse of the resistance) of the sensor.
- Oxidation-Reduction Potential (ORP) sensor measures the output voltage between a measuring electrode and a reference electrode, which indicates the ability of a water body to acquire electrons, thereby to be reduced.

The WMSN consists of a group of USVs. The designed framework and the developed USV are shown in Figure 6.2 and Figure 6.3, respectively. Each node consists of five sensors, a control unit, a data processing unit, and two power supply modules. To avoid the inaccurate measurements caused by surrounding objects (the magnetic field between the electrodes may be affected by the surroundings), the five sensors are held separately through a Polyvinyl Chloride (PVC) structure. All electronic components are

deployed inside a waterproof floating buoy. The conversion of the sensor output signals (e.g., voltages) to the sensor readings, which indicate the real concentrations, is carried out by a mote, Waspmote with microcontroller ATmega1281, which is an advanced mote manufactured by Libelium Communicaciones Distribuidas S.L. Data is processed at the on-board processor, Raspberry Pi 3, and then transmitted to the BS through a Wi-Fi or Zigbee radio transmitter. To enable mobility, a BlueRobotics ROV external structure with two T200 propellers are integrated. A 3DR Pixhawk mini with a GPS module is equipped as the autopilot of the MSN. To avoid data missing due to package loss or communication failure during data transmission, recently collected data is stored at the local data logger. A 3.7 V, 6600 mA·h LiPo battery is used to provide power for sensing and wireless transmission. It also stores the harvested energy from a 23·16·20 mm solar panel. In addition, a 14.8 V, 10 A·h LiPo battery is used to supply power for the autopilot, the Electronic Speed Controllers (ESCs) and the propellers.



Figure 6.2: The design framework of the MSN.



Figure 6.3: The developed MSN in the wireless mobile sensor network.

6.3.2 Base Station and Remote Server

The design framework and the developed BS are shown in Figure 6.4 and Figure 6.5, respectively. It consists of a gateway, an LAU, and a power supply module. The gateway, the Meshlium Xtreme manufactured by Libelium, is deployed in the BS. It is a Linux-based router, which works as the gateway for the local wireless network. The radio receiver of the gateway receives the data transmitted from the distributed MSNs. The data received from the MSNs is stored in a MySQL database embedded in the gateway. The quality indexing algorithm is operated at the LAU. The monitoring results can be accessed at the BS by a laptop (i.e., an LAU) through a local Graphic User Interface (GUI), mainly for local examination by technicians in the field. The data is then transmitted to the RS as well. The gateway accesses the cellular network to communicate with the RS via the Internet. The solar energy is collected by a 48·43·30 mm solar panel and stored into a 12 V DC battery through a solar energy charge controller. Then 110 V AC power is supplied to the gateway through a DC/AC power inverter, which is connected to the charge controller.

The RS is a PC with a 4.00 GHz Intel Core i7-6700K CPU and a 32 GB RAM. By implementing the proposed survey planner, the sensing missions including the sampling locations and the paths for USVs are generated by a CAU at the RS. The missions are then transmitted to the USVs via the BS. The RS also receives the data collected in the field and interprets the data online for users.



Figure 6.4: The design architecture of the BS.



Figure 6.5: The developed BS in the sensing platform.

6.4 Case Study

To evaluate the proposed on-line index and the developed platform, experiments are conducted systematically using a real-world dataset and through physical field testing. The experiment using the dataset examines the proposed on-line indexing approach in relation to the statistical performance of the index factors. In the field tests, the overall tasks of the platform are evaluated, including sampling design, path planning, on-line indexing, and system implementation.

6.4.1 Real-World Dataset

For validation of the performance of online indexing, both the CCME WQI and the OLWQI are applied to a realistic dataset with a large quantity of continuous online measurements that have been collected by the Chesapeake Bay Interpretive Buoy System (CBIBS) [4]. The data from 1 June, 2017 to 1 July, 2017 at 10 monitoring stations in the dataset are selected for the evaluation.

The measurements of six parameters that are related to water quality (namely, Chlorophyll A, Dissolved Oxygen, Turbidity, Conductivity, Salinity, and Temperature) are used in the index calculation. The index is calculated hourly by aggregating the latest 24-hour data for quality evaluation (24-hour sliding time window). Due to the limited space, Table 6.3 shows some of the indexing results by applying the CCME WQI and the proposed OLWQI on the dataset.

Date & Time	F_1	F_2	F_3	CCME WQI	F_1'	F_2'	F'_3	OLWQI
6/20/2017 13:00	16.67	6.94	0.92	90	0	6.94	7.46	94
6/20/2017 14:00	16.67	6.94	0.92	90	0	6.94	7.46	94
6/20/2017 15:00	16.67	6.94	0.92	90	0	6.94	7.46	94
6/20/2017 16:00	16.67	7.64	0.99	89	0	7.64	8.01	94
6/20/2017 17:00	16.67	8.33	1.20	89	16.67	8.33	9.59	88
6/20/2017 18:00	16.67	9.03	1.50	89	16.67	9.03	11.75	87
6/20/2017 19:00	16.67	9.03	1.52	89	16.67	9.03	11.89	87
6/20/2017 20:00	16.67	9.03	1.52	89	16.67	9.03	11.89	87
6/20/2017 21:00	16.67	9.03	1.52	89	16.67	9.03	11.86	87
6/20/2017 22:00	16.67	9.03	1.53	89	16.67	9.03	11.95	87

Table 6.3: Indexing results using data collected at the First Landing Station in CBIBS.

To demonstrate the resulting factor effects on the final index score, we have statistically summarized both indices by applying stepwise regression analysis. Two examples are given in Table 6.4 to present the statistical summary of the final index versus the three factors utilizing the one-month data from the single station (First Landing Station), and the data from the regional multiple stations (First Landing Station and York Spit Station). Both examples in the table demonstrate that the Scope Factor in the CCME WQI dominates the whole index, as indicated by the adjusted R-squared value of step 1. Meanwhile, the Amplitude Factor in the CCME WQI has very limited influence on the final index score, as indicated by the adjusted R-squared value of step 3.

In this example, the stepwise regression analysis has been executed 20 times on the selected dataset, including 10 times on the 10 single stations, and 20 times on the pairs of neighboring stations. The statistical results are summarized in Table 6.5. In the table, $f_{F_i} = \sum_{j=1}^{k} R_{ij} / K$, where k = number of times that the factor F_i was obtained in the first step, i = 1, 2, 3; R_{ij} = corresponding adjusted R-squared value in the first step of the factor F_i ; and K = total number of experimental runs. The results demonstrate that the proposed OLWQI provides indexing results with a more balanced factor sensitivity for large quantities of online data that are automatically collected, compared to the CCME WQI.

		CCM	IE WQI			O	LWQI	
Single Station	Candidate	Step 1	Step 2	Step 3	Candidate	Step 1	Step 2	Step 3
	Term	(Coef)	(Coef)	(Coef)	Term	(Coef)	(Coef)	(Coef)
	F_1	-0.5808	-0.5742	-0.5744	F_1'	-0.8511	-0.2145	-0.2378
	F_2		-0.0690	-0.0611	F_2'		-0.9327	-0.4381
	F_3			-0.0520	F'_3			-0.3588
	R ² (adj)	98.00%	99.91%	99.92%	R ² (adj)	83.97%	98.75%	99.81%
	Candidate	Step 1	Step 2	Step 3	Candidate	Step 1	Step 2	Step 3
	Term	(Coef)	(Coef)	(Coef)	Calificate	(Coef)	(Coef)	(Coef)
	F_1	-0.5966	-0.5502	-0.5528	F_1'	-0.8347	-0.3192	-0.3002
	F_2		-0.1821	-0.1536	F_2'		-1.4212	-0.3049
	F_3			-0.0690	F_3'			-0.4066
	R ² (adj)	98.47%	99.94%	99.95%	R ² (adj)	85.43%	97.95%	99.89%

Table 6.4: Stepwise regression analysis: final index versus three factors.

CCME WQI			OLWQI			
f_{F_1}	f_{F_2}	f_{F_3}	$f_{F_1'}$	$f_{F_2'}$	$f_{F_3'}$	
79.89%	3.64%	7.57%	15.75%	36.98%	37.88%	

Table 6.5: Statistical results of the factor effects on the final index.

6.4.2 Field Test

The developed sensing platform has been deployed at the Yosef Wosk Reflecting Pool of the University of British Columbia, Canada. The in situ deployment of the platform is shown in Figure 6.6.



(a)

(b)

Figure 6.6: Field deployment of the platform at the Yosef Wosk Reflecting Pool. (a) Deployment of the USV in the pool; (b) Deployment of the platform in the field.

The platform core software has been developed in Java and executed on the RS to implement the proposed HGCSP algorithm and the OLWQI algorithm. The geometric map of the study area is embedded in the software. The survey mission is generated at the core and then sent to the USVs via the BS. The robotic operating system (ROS) is executed on the Raspberry Pi 3 in the USVs to control and navigate the MSN to move to the objective sampling locations that are included in the survey mission. The collected data are transmitted to the RS via the BS and stored in the MySQL database. The core software then processes the collected data to calculate the OLWQI.

Before the USVs are deployed in the water body, all the sensors in each robot are calibrated carefully in the field to assure good data sample quality in the initial stages. In addition, two MSNs are fully charged and launched at the predetermined location. In the field test, the coordinates of the SLoIs are translated from the meter scale to GPS coordinates at the RS, and uploaded as the waypoints to the USVs via the BS. The two USVs move along the target coverage path four times, while each sampling location is measured eight times over two hours. Figure 6.7 shows a GUI display of the experiment scenario. The path is planned based on the energy budget and the sampling frequency requirement, which are set using the "Generate Survey Plan" function in the GUI. The resulting planned path is displayed in the "Plan View." The blue circles in the plan view denote the division points for the sub-paths, and the hollow one indicates the initial launching position of the MSNs. After successfully uploading the survey mission to the MSNs, the in situ data sampling locations of the MSNs can be checked during the surveying process through the "Survey Process View" in the GUI.



Figure 6.7: Graphical User Interface (GUI).

In the experiment, data samples collected from the field test are used to demonstrate the performance of the OLWQI in comparison to the state-of-the-art CCME WQI. The proposed OLWQI is primarily intended for short-term water quality evaluation using data that are collected through automated sampling. The category interpretation based on the index score is given by, Excellent: 95–100; Good: 80–94; Fair: 65–79; Marginal: 45–64; Poor: 0-44 [71]. The algorithm for calculating the OLWQI is implemented in the LAU and CAU, where the temporal and spatial scales can be set in a flexible manner by the user. In the experiments, a one-hour window is used to calculate the OLWQI (four sampling cycles are involved) in a temporal point of view. Meanwhile, the quality index of each cell, the overall study area, and a selected objective area are provided and displayed in the GUI (see Figure 6.7). The quality indexing results at all the fine hexagonal cells are displayed in the "Quality Index View," indicated by five different colors, which denote the five quality categories. The overall OLWQI are calculated by the integration of the data collected at all the SLoIs over the study area. By utilizing the "Select Objective Area" function in the GUI, the users are able to check the indexing results across any objective area that is selected in the view.

To demonstrate the performance of the OLWQI in comparison with the CCME WQI, both indices have been implemented in the experiment. In Figure 6.8, data samples are selected from a one-hour time window to illustrate the indexing results of the overall study area (overall WQI) by using these two indices. In the time window, seven tests of pH value and five tests of EC have failed (readings do not meet their relative guidelines, pH: 6.5-9, EC: <2000 μ S/cm) among the total 1760 tests. This means 0.68% tests failed in all tests. As shown in Figure 6.8(a), the overall index score of the CCME WQI is 77/100 (Fair). The Scope Factor has dominated the total index score of the CCME WQI. Particularly, in Figure 6.8(b), the Scope Factor in the CCME WQI has degraded by 40 units due to these few failed tests, leading to the unreliable index score of the CCME WQI. In fact, in the worst-case scenario, only two failed tests from these two parameters (pH value and EC) can cause such a 40-unit degradation. By contrast, the OLWQI indicates an online indexing result of 98/100 (Excellent). It follows that the Scope Factor in the OLWQI has not been affected unreasonably by the small number of failed tests.



Figure 6.8: Comparison of the CCME WQI and the OLWQI. (a) Quality indexing results; (b) Factor scores due to the failed tests.

The influence of the Amplitude Factor can be ignored if the CCME WQI is implemented on a large volume of online sampled data. This issue is demonstrated in Figure 6.8. The modification in the OLWQI for the Amplitude Factor makes it effective for a large volume of data. In addition, the water quality parameters with different data ranges result in different excursion ranges for the Amplitude Factor when the CCME WQI is implemented. Notice that the parameter with a large value range (e.g., EC) results in a large excursion range. This may lead to a biased index score. For example, a test of EC = 3200 μ S/cm leads to the excursion of 0.6 in the Amplitude Factor of the CCME WQI. It even exceeds the upper limit of the pH excursion, since the maximum pH value is 14 (maximum pH excursion: 14/9-1=0.56). Hence, a parameter may dominate the Amplitude Factor in the CCME WQI. In contrast, the OLWQI provides the same excursion range (from 0 to 1) for different parameters, regardless of their data range and guidelines.

6.5 Summary

This chapter presented an on-line sensing platform, which has been developed in the present work, for surface water monitoring. The developed platform provided a cost-effective, fast, deployable, and easily maintainable solution for the high-resolution spatiotemporal mobile sensing. The developed sensing platform has been deployed in a real water source to demonstrate the implementation and performance of the overall system.

A novel online water quality index, the OLWQI, was developed based on the stateof-the-art water quality index, CCME WQI, to evaluate water quality by integrating a large volume of online data acquired through automated sampling. The OLWQI that was formulated in the analytical form facilitated online processing by automatic execution on automated devices. The performance of the OLWQI in compassion to the existing CCME WQI was demonstrated by using data from the CBIBS dataset and realistic data that was collected through field experiments using our developed platform. The experimental results show that the proposed OLWQI provided rather balanced factor sensitivities and more reliable evaluation results for online quality indexing compared to the CCME WQI.

Chapter 7: Conclusions and Future Work

In this chapter, the primary research goals and contributions of the present dissertation are summarized. The promising directions and recommendations are given for possible future research.

7.1 Conclusions

In this dissertation, several research and development issues of applying mobile sensing for automated environmental field estimation were investigated. In particular, exploratory sampling, coverage path planning, environmental model estimation, spatial field mapping, informative path planning, adaptive sampling, and on-line quality indexing were addressed. This dissertation made direct contributions in these issues while developing a system framework and application platform with novel schemes and strategies. The present work particularly focused on the robotic sensing applications in automated aquatic environmental monitoring.

In order to generate a coverage sampling frame and a coverage sampling path to carry out an initial exploratory survey, the problem of the regular grid-based survey planning was investigated. First, a hexagonal grid-based frame for coverage sampling was designed to plot spatially balanced sampling locations across the entire field for a sampling resolution of interest. Second, an energy-efficient coverage path cycle was planned to travel among the objective locations for automated data collection. A hexagonal grid-based spanning tree-assisted coverage (HGSTC) survey planner was developed to construct the coverage path by circumnavigating the constructed MST. A hexagonal grid-based coverage (HGC) survey planner was proposed to further improve the computational efficiency based on the proposed ACC decomposition method. The developed algorithms targeted the mission planning of mobile sensing robots for the prior exploratory survey phase in environmental monitoring.

In real-world robotic sensing applications, power supply and energy budget are strictly constrained in the systems, which is a crucial limitation in applying mobile sensing robots. An energy-constrained survey planner was proposed to find the optimal coverage density for exploratory sampling, which guaranteed that the corresponding energy consumption in carrying out the planned sampling path was within the given energy budget. The GP model and the Kriging approach were implemented to establish the underlying environmental model and map the spatial field. The performance of model estimation, field mapping, and computational expense determination were carried out by the proposed algorithm and results were compared with the state-of-the-art approaches. The proposed planner demonstrated its superior performance for automated and energyconstrained survey planning in field estimation and mapping of an unknown environment.

The proposed planners generated the shortest path to visit the SLoIs under the hexagonal tessellation for a given grid size. Note that the SLoIs were designed according to the corresponding grid size. For a certain region with regular or irregular boundary, the proposed planners determined the maximum sampling sites that an exploratory survey can achieve, given the number and energy budgets of the robots. It means the minimum size of the grid size is bounded and determined by the robot number and the power supply conditions of the robots. Furthermore, the planned survey path could be assigned to multiple robots based on the requirement of the time interval for measuring a sampling site.

With accumulated knowledge from initial and historical deployments, adaptive sensing that focuses on measuring locations that are more significant is able to gather information more efficiently. To this end, the sites that are most informative among the overall potential positions should be chosen as the sampling targets. A hierarchical informative path planner was proposed for navigating the mobile sensing robots to the most informative sites, which were evaluated by the information-theoretic metric MI. The GMRF model was built in the planning framework to approximate the GP model for computational feasibility. Beneficial characteristics of the proposed hierarchical framework assured the balance between the intractable global optimization for optimal design and the fast local greedy search for on-line path planning. The proposed planner provided an effective and efficient solution for long-term spatiotemporal monitoring.

In addition to the quantitative evaluation, qualitative evaluation of the monitored area was studied in the present work. A novel quality indexing approach, OLWQI, was developed to integrate and interpret the quality information from a large volume of on-line measurements. The OLWQI provided more balanced factor sensitivities compared to the popular and traditional indexing approach, CCME WQI. It was formulated in an analytical form, which facilitated online processing in automated execution. The design and development of a WMSN for surface water monitoring was introduced. The sensing platform has been deployed in a real-world aquatic environment. The field experiments demonstrated the functionality and utility of the developed WMSN in on-line monitoring of surface water. The results of the numerical analysis of practical data and the field tests demonstrated the effectiveness and feasibility of the proposed planning and indexing schemes as well as the practical implementation of the automated system.

7.2 Possible Future Work

Future research may focus on several promising directions. In the present dissertation, a two-dimensional planar was considered as the environmental field of interest. The proposed algorithms were conducted on a two-dimensional spatial field. The future work may modify and extend them for a three-dimensional space while incorporating both spatial and temporal scales. In this manner, the planners developed in the present work can be implemented in more general and extensive robotic sensing scenarios such as marine monitoring with Autonomous Underwater Vehicles (AUV). Moreover, other environmental fields can also be monitored by applying the proposed planning approaches. For example, air quality of an atmospheric environment can be sensed with Unmanned Aerial Vehicles (UAV) by making use of the planners in the present work. Since the UAVs generally have limited on-board resources, the on-line planning issues that are addressed in the present work will benefit the UAV-based sensing applications in air monitoring.

The GP and GMRF models that were integrated in the present dissertation have been developed in a solid mathematical framework to deal with stationary random field. However, a more complicated environmental model may be required to describe, estimate, and interpret some real-world natural phenomenon. They may need to be modeled by a non-stationary GP model or even a non-GP model. The planning schemes developed in the present work may be further investigated based on more complicated environmental models.

Future research may also concentrate on the variation of the established statistical field model over the temporal scale. One direct way is to re-implement the proposed

exploratory sampling strategies to learn a new underlying environment. However, using accumulated knowledge from historical measurements, more efficient planning strategies can be incorporated. For example, proper balance between coverage sampling and adaptive sampling may be targeted to tackle the exploration-exploitation problem in a changing environment. This point may be undertaken as a desirable future research direction.

Another possible research direction is the application of deep learning techniques to exploit underlying environmental models using the titanic amounts of historical data. The statistical model of some natural phenomenon may not be readily or even possibly represented in an analytical structure. With the fast development of a mathematical framework and hardware support, deep learning techniques such as convolutional neural networks (CNN) demonstrate their superior ability in representative learning and information exploitation. Accordingly, leaning a complicated environment model and selecting the essential sampling sites through deep learning techniques may be investigated to guide robotic path planning for environmental monitoring.

In conclusion, the WMSN developed in the present work may be tested in a more complicated aquatic environment and for long-term implementation. Many factors may affect the effectiveness of the developed methods. For example, complex and high water turbulences make an influence on the mobile sensing robots to carry out their sampling missions. A local motion plan may be integrated to handle possible bad dynamic conditions of a water body. Furthermore, if the motion control is not able to guide the robots to visit the waypoints of the sampling missions. A more robust scheme may be proposed as a backup plan to complete the sampling missions. Also, location uncertainties of the mobile robots may be addressed. The proposed sensing, planning, and indexing approaches can be further examined in the future field experiments.

References

- C. Alippi, R. Camplani, C. Galperti, and M. Roveri, "A robust, adaptive, solarpowered WSN framework for aquatic environmental monitoring," *IEEE Sensors Journal*, vol. 11, no. 1, pp. 45–55, 2011.
- [2] K. Sarpong Adu-manu, C. Tapparello, W. Heinzelman, F. Apietu Katsriku, and J. Abdulai, "Water quality monitoring using wireless sensor networks: current trends and future research directions," *ACM Transactions on Sensor Networks*, vol. 13, no. 1, pp. 1–41, 2017.
- [3] Environment Canada, "Canada environment Fraser river water quality buoy."
 [Online]. Available: http://aquatic.pyr.ec.gc.ca/Real TimeBuoys/Default.aspx.
 [Accessed: 18-Jul-2017].
- [4] NOAA, "Chesapeake Bay Interpretive Buoy System." [Online]. Available: https://buoybay.noaa.gov/. [Accessed: 01-Jun-2018].
- [5] G. S. Sukhatme, A. Dhariwal, B. Zhang, C. Oberg, B. Stauffer, and D. A. Caron, "Design and development of a wireless robotic networked aquatic microbial observing system," *Environmental Engineering Science*, vol. 24, no. 2, pp. 205–215, 2007.
- [6] T. Taher, G. D. Weymouth, and T. Varghese, "Novel platform for ocean survey and autonomous sampling using multi-agent system," in *Proceedings of the 2013 MTS/IEEE Oceans Conference*, 2013, pp. 1–5.
- [7] J. Laut, E. Henry, O. Nov, and M. Porfiri, "Development of a mechatronics-based citizen science platform for aquatic environmental monitoring," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 5, pp. 1541–1551, 2014.
- [8] Y. Wang, R. Tan, G. Xing, X. Tan, J. Wang, and R. Zhou, "Spatiotemporal aquatic field reconstruction using cyber-physical robotic sensor systems," ACM *Transactions on Sensor Networks*, vol. 10, no. 4, p. 57, 2014.
- [9] A. Mora, C. Ho, and S. Saripalli, "Analysis of adaptive sampling techniques for underwater vehicles," *Autonomous Robots*, vol. 35, no. 2–3, pp. 111–122, 2013.
- [10] T. P. Lambrou and C. G. Panayiotou, "Collaborative path planning for event search and exploration in mixed sensor networks," *International Journal of Robotics*

Research, vol. 32, no. 12, pp. 1424–1437, 2013.

- [11] S. A. Sadat, J. Wawerla, and R. Vaughan, "Fractal trajectories for online nonuniform aerial coverage," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2015, pp. 2971–2976.
- [12] N. Xia, C. Wang, Y. Yu, H. Du, C. Xu, and J. Zheng, "A path forming method for water surface mobile sink using Voronoi diagram and dominating set," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 7608–7619, 2018.
- [13] Y. Liu and R. Bucknall, "Efficient multi-task allocation and path planning for unmanned surface vehicle in support of ocean operations," *Neurocomputing*, vol. 275, pp. 1550–1566, 2018.
- [14] R. Graham and J. Cortes, "Adaptive information collection by robotic sensor networks for spatial estimation," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1404–1419, 2012.
- [15] W. C. Evans, D. Dias, S. Roelofsen, and A. Martinoli, "Environmental field estimation with hybrid-mobility sensor networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2016, pp. 5301–5308.
- [16] W. Xu, P. Collingsworth, B. Bailey, M. Carlson Mazur, J. Schaeffer, and B. Minsker, "Detecting spatial patterns of rivermouth processes using a geostatistical framework for near-real-time analysis," *Environmental Modelling and Software*, vol. 97, pp. 72–85, 2017.
- [17] K.-C. Ma, L. Liu, H. K. Heidarsson, and G. S. Sukhatme, "Data-driven learning and planning for environmental sampling," *Journal of Environmental Engineering*, vol. 35, no. 5, pp. 643–661, 2017.
- [18] T. Somers and G. A. Hollinger, "Human–robot planning and learning for marine data collection," *Autonomous Robots*, vol. 40, no. 7, pp. 1123–1137, 2016.
- [19] M. Dunbabin and L. Marques, "Robots for environmental monitoring: Significant advancements and applications," *IEEE Robotics and Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.
- [20] Y. Xu, J. Choi, S. Dass, and T. Maiti, Bayesian Prediction and Adaptive Sampling Algorithms for Mobile Sensor Networks: Online Environmental Field Reconstruction in Space and Time. Springer, 2015.

- [21] A. Lumb, T. C. Sharma, and J.-F. Bibeault, "A review of genesis and evolution of water quality index (WQI) and some future directions," *Water Quality, Exposure and Health*, vol. 3, no. 1, pp. 11–24, 2011.
- [22] X. Huang, J. Yi, S. Chen, and X. Zhu, "A wireless sensor network-based approach with decision support for monitoring lake water quality," *Sensors*, vol. 15, pp. 29273–29296, 2015.
- [23] F. Zhang, O. Ennasr, E. Litchman, and X. Tan, "Autonomous sampling of water columns using gliding robotic fish: algorithms and harmful-algae-sampling experiments," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1271–1281, 2016.
- [24] J. M. Dolan *et al.*, "Cooperative aquatic sensing using the telesupervised adaptive ocean sensor fleet," in *In Proceedings of the SPIE Europe Remote Sensing*, *International Society for Optics and Photonics*, 2009, vol. 7473, pp. 747307-1-747307-12.
- [25] C. Koparan, A. B. Koc, C. V Privette, and C. B. Sawyer, "In situ water quality measurements using an unmanned aerial vehicle (UAV) system," *Water*, vol. 10, no. 3, p. 264, 2018.
- [26] D. J. Brus, "Balanced sampling: A versatile sampling approach for statistical soil surveys," *Geoderma*, vol. 253–254, pp. 111–121, 2015.
- [27] J. De Gruijter, D. J. Brus, M. F. Bierkens, and M. Knotters, *Sampling for Natural Resource Monitoring*. Springer Science & Business Media, 2006.
- [28] G. H. Strand, "A study of variance estimation methods for systematic spatial sampling," *Spatial Statistics*, vol. 21, pp. 226–240, 2017.
- [29] J. J. De Gruijter, B. Minasny, and A. B. Mcbratney, "Optimizing stratification and allocation for design-based estimation of spatial means using predictions with error," *Journal of Survey Statistics and Methodology*, vol. 3, no. 1, pp. 19–42, 2015.
- [30] A. BISWAS and Y. ZHANG, "Sampling designs for validating digital soil maps: A review," *Pedosphere*, vol. 28, no. 1, pp. 1–15, 2018.
- [31] R. Webster and R. M. Lark, *Field Sampling for Environmental Science and Management*. Routledge, 2012.
- [32] M. J. De Smith, M. F. Goodchild, and P. Longley, *Geospatial Analysis: A Comprehensive Guide to Principles, Techniques and Software Tools (5th ed).*

Winchelsea Press, 2015.

- [33] D. J. J. Walvoort, D. J. Brus, and J. J. de Gruijter, "An R package for spatial coverage sampling and random sampling from compact geographical strata by k-means," *Computers and Geosciences*, vol. 36, no. 10, pp. 1261–1267, 2010.
- [34] Y. Ge, J. H. Wang, G. B. M. Heuvelink, R. Jin, X. Li, and J. F. Wang, "Sampling design optimization of a wireless sensor network for monitoring ecohydrological processes in the Babao River basin, China," *International Journal of Geographical Information Science*, vol. 29, no. 1, pp. 92–110, 2015.
- [35] R. M. Lark, E. M. Hamilton, B. Kaninga, K. K. Maseka, M. Mutondo, G. M. Sakala, and M. J. Watts, "Planning spatial sampling of the soil from an uncertain reconnaissance variogram," *Soil*, vol. 3, no. 4, pp. 235–244, 2017.
- [36] R. M. Lark and B. P. Marchant, "How should a spatial-coverage sample design for a geostatistical soil survey be supplemented to support estimation of spatial covariance parameters?," *Geoderma*, vol. 319, no. January, pp. 89–99, 2018.
- [37] J. F. Wang, A. Stein, B. B. Gao, and Y. Ge, "A review of spatial sampling," *Spatial Statistics*, vol. 2, no. 1, pp. 1–14, 2012.
- [38] A. J. Lister and C. T. Scott, "Use of space-filling curves to select sample locations in natural resource monitoring studies," *Environmental Monitoring and Assessment*, vol. 149, no. 1–4, pp. 71–80, 2009.
- [39] W. G. Müller, Collecting Spatial Data: Optimum Design of Experiments for Random Fields. Springer Science & Business Media, 2007.
- [40] L. Paull, S. Saeedi, M. Seto, and H. Li, "Sensor-driven online coverage planning for autonomous underwater vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1827–1838, 2013.
- [41] R. A. Viscarra Rossel, R. Webster, and D. Kidd, "Mapping gamma radiation and its uncertainty from weathering products in a Tasmanian landscape with a proximal sensor and random forest kriging," *Earth Surface Processes and Landforms*, vol. 39, no. 6, pp. 735–748, 2014.
- [42] L. David, R. E. Bixby, V. Chvatal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2006.
- [43] J. Yu, J. Aslam, S. Karaman, and D. Rus, "Anytime planning of optimal schedules

for a mobile sensing robot," in *Proceedings of the IEEE International Conference* on Intelligent Robots and Systems, 2015, pp. 5279–5286.

- [44] G. Gutin and (Eds.) Abraham P. Punnen, *The Traveling Salesman Problem and its Variations*. Springer Science & Business Media, 2006.
- [45] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.
- [46] C. Fang and S. Anstee, "Coverage path planning for harbour seabed surveys using an autonomous underwater vehicle," in *Proceedings of the 2010 MTS/ IEEE Oceans Conference*, 2010, pp. 1–8.
- [47] T. Wilson and S. B. Williams, "Adaptive path planning for depth-constrained bathymetric mapping with an autonomous surface vessel," *Journal of Field Robotics*, no. March 2016, pp. 345–358, 2017.
- [48] M. S. Stanković and D. M. Stipanović, "Extremum seeking under stochastic noise and applications to mobile sensors," *Automatica*, vol. 46, no. 8, pp. 1243–1251, 2010.
- [49] H. M. La and W. Sheng, "Distributed sensor fusion for scalar field mapping using mobile sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 43, no. 2, pp. 766–778, 2013.
- [50] X. Lan and M. Schwager, "Rapidly exploring random cycles: Persistent estimation of spatiotemporal fields with multiple sensing robots," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1230–1244, 2016.
- [51] L. V Nguyen, S. Kodagoda, R. Ranasinghe, and G. Dissanayake, "Informationdriven adaptive sampling strategy for mobile robotic wireless sensor network," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 1, pp. 372–379, 2016.
- [52] E. M. Hanks, E. M. Schliep, M. B. Hooten, and J. A. Hoeting, "Restricted spatial regression in practice: Geostatistical models, confounding, and robustness under model misspecification," *Environmetrics*, vol. 26, no. 4, pp. 243–254, 2015.
- [53] J. Montero, G. Fernández-Avilés, and J. Mateu, *Spatial and Spatio-temporal Geostatistical Modeling and Kriging*, vol. 998. John Wiley & Sons, 2015.
- [54] M. A. Oliver and R. Webster, Basic Steps in Geostatistics: The Variogram and

Kriging, vol. 106. New York: Springer, 2015.

- [55] N. Cressie, *Statistics for Spatial Data*. John Wiley & Sons, 2015.
- [56] H. Rue, *Gaussian Markov Random Fields: Theory and Applications*. CRC press, 2005.
- [57] F. Lindgren, H. Rue, and J. Lindström, "An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach," *Journal of the Royal Statistical Society, Series B*, vol. 73, no. 4, pp. 423– 498, 2011.
- [58] V. V. Fedorov and P. Hackl., *Model-oriented Design of Experiments*, vol. 125. Springer Science & Business Media, 2012.
- [59] S. P. Chepuri and G. Leus, "Sparsity-promoting sensor selection for non-linear measurement models," *IEEE Transactions on Signal Processing*, vol. 63, no. 3, pp. 684–698, 2015.
- [60] V. Roy, A. Simonetto, and G. Leus, "Spatio-temporal sensor management for environmental field estimation," *Signal Processing*, vol. 128, pp. 369–381, 2016.
- [61] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [62] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.
- [63] H. Boyacioglu, "Utilization of the water quality index method as a classification tool," *Environmental Monitoring and Assessment*, vol. 167, no. 1–4, pp. 115–124, 2010.
- [64] R. K. Horton, "An index number system for rating water quality," *Journal of Water Pollution Control Federation*, vol. 37, no. 3, pp. 300–306, 1965.
- [65] A. Lumb, T. C. Sharma, and J.-F. Bibeault, "A review of genesis and evolution of Water Quality Index (WQI) and some future directions," *Water Quality, Exposure* and Health, 2011.
- [66] R. M. Brown, N. I. McClelland, R. A. Deininger, and R. Tozer, "A Water quality index-Do we dare," *Water Sewage Works*, vol. 117, no. 10, pp. 339–343, 1970.
- [67] D. S. Bhargave, "Expression for drinking water supply standards," Journal of

Environmental Engineering, vol. 111, no. 3, pp. 304–316, 1985.

- [68] C. G. Cude, "Oregon water quality index: A tool for evaluating water quality management effectiveness," *Journal of the American Water Resources Association*, vol. 37, no. 1, pp. 125–137, 2001.
- [69] CCME, "Canadian water quality guidelines for the protection of aquatic life: CCME water quality index 1.0, technical report," 2001.
- [70] CCME, "Canadian water quality guidelines for the protection of aquatic life: CCME water quality index 1.0, user's manual," 2001.
- [71] CCME, "Canadian water quality guidelines for the protection of aquatic life: CCME water quality index, user's manual 2017 update," 2017.
- [72] A. Lumb, D. Halliewell, and T. Sharma, "Application of CCME water quality index to monitor water quality: A case of the Mackenzie River Basin, Canada," *Environmental Monitoring and Assessment*, vol. 113, no. 1–3, pp. 411–429, 2006.
- [73] UNEP, "Global drinking water quality index development and sensitivity analysis report," 2007.
- [74] T. Hurley, R. Sadiq, and A. Mazumder, "Adaptation and evaluation of the Canadian Council of Ministers of the Environment Water Quality Index (CCME WQI) for use as an effective tool to characterize drinking source water quality," *Water Research*, vol. 46, no. 11, pp. 3544–3552, 2012.
- [75] G. da Costa Silva and M. G. Dubé, "Water quality assessment at a global scale: A comparison between world regions," *Water International*, vol. 38, no. 1, pp. 78–94, 2013.
- [76] Y. Gabriely and E. Rimon, "Spanning-tree based coverage of continuous areas by a mobile robot," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1–4, pp. 77–98, 2001.
- [77] N. Hazon and G. A. Kaminka, "On redundancy, efficiency, and robustness in coverage for multiple robots," *Robotics and Autonomous Systems*, vol. 56, no. 12, pp. 1102–1114, 2008.
- [78] K. R. Guruprasad and T. D. Ranjitha, "ST-CTC : a spanning tree-based competitive and truly complete coverage algorithm for mobile robots," in *Proceedings of the* 2015 ACM Conference on Advances In Robotics, 2015, p. 43.

- [79] V. S. Gordon, Y. L. Orlovich, and F. Werner, "Hamiltonian properties of triangular grid graphs," *Discrete Mathematics*, vol. 308, no. 24, pp. 6166–6188, 2008.
- [80] E. M. Arkin *et al.*, "Not being (super)thin or solid is hard: A study of grid Hamiltonicity," *Computational Geometry: Theory and Applications*, vol. 42, no. 6– 7, pp. 582–605, 2009.
- [81] K. Hormann and A. Agathos, "The point in polygon problem for arbitrary polygons," *Computational Geometry: Theory and Applications*, vol. 20, no. 3, pp. 131–144, 2001.
- [82] V. R. Joseph, Y. Hung, and A. Sudjianto, "Blind Kriging: A new method for developing metamodels," *Journal of Mechanical Design*, vol. 130, no. 3, p. 031102, 2008.
- [83] I. Couckuyt, A. Forrester, D. Gorissen, F. De Turck, and T. Dhaene, "Blind Kriging: Implementation and performance analysis," *Advances in Engineering Software*, vol. 49, no. 1, pp. 1–13, 2012.
- [84] C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.
- [85] A. C. Kapoutsis, S. A. Chatzichristofis, and E. B. Kosmatopoulos, "DARP: Divide areas algorithm for optimal multi-robot coverage path planning," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 86, no. 3–4, pp. 663– 680, 2017.
- [86] NOAA, "Noaa operational model archive and distribution system (oceannomads),"
 2018. [Online]. Available: url: http://ecowatch.ncddc.noaa.gov/.
- [87] Y. Xu and J. Choi, "Adaptive sampling for learning Gaussian processes using mobile sensor networks," *Sensors*, vol. 11, no. 3, pp. 3051–3066, 2011.
- [88] H. Rue and H. Tjelmeland, "Fitting Gaussian Markov random fields to Gaussian fields," *Scandinavian Journal of Statistics*, vol. 29, no. 1, pp. 31–49, 2002.
- [89] Y. Xu, J. Choi, S. Dass, and T. Maiti, "Efficient Bayesian spatial prediction with mobile sensor networks using Gaussian Markov random fields," *Automatica*, vol. 49, no. 12, pp. 3520–3530, 2013.
- [90] L. V Nguyen, S. Kodagoda, and R. Ranasinghe, "Spatial sensor selection via gaussian markov random fields," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 9, pp. 1226–1239, 2016.

- [91] L. V Nguyen, S. Kodagoda, R. Ranasinghe, and G. Dissanayake, "Mobile robotic wireless sensor networks for efficient spatial prediction," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2014, pp. 1176– 1181.
- [92] F. Lindgren and H. Rue, "Bayesian spatial modelling with R-INLA," Journal of Statistical Software, vol. 63, no. 19, 2015.
- [93] N. L. Ramanathan Balzano M Burt *et al.*, "Rapid deployment with confidence: calibration and fault detection in environmental sensor networks," 2006.
- [94] T. Hurley, R. Sadiq, and A. Mazumder, "Adaptation and evaluation of the Canadian Council of Ministers of the Environment Water Quality Index (CCME WQI) for use as an effective tool to characterize drinking source water quality," *Water Research*, vol. 46, no. 11, pp. 3544–3552, 2012.
- [95] M. Terrado, D. Barceló, R. Tauler, E. Borrell, and S. de Campos, "Surface-waterquality indices for the analysis of data generated by automated sampling networks," *TrAC - Trends in Analytical Chemistry*, vol. 29, no. 1, pp. 40–52, 2010.
- [96] Canadian Council of Minister of the Environment, "A protocol for the derivation of water quality guidelines for the protection of aquatic life 2007," 2007.
- [97] A. Lee, A. Francisque, H. Najjaran, M. J. Rodriguez, M. Hoorfar, S. A. Imran, and R. Sadiq, "Online monitoring of drinking water quality in a distribution network: a selection procedure for suitable water quality parameters and sensor devices," *International Journal of System Assurance Engineering and Management*, vol. 3, no. 4, pp. 323–337, 2012.

Appendix

Appendix A

A.1 **Proof of Proposition 5.3.**

Consider $\omega \in [0, \omega_{\max} = \arcsin(2\sin\frac{\theta}{2})], \ \gamma \in [-\theta, \theta], \ \theta \in [0, \frac{\pi}{4}]$, the distance from \mathbf{v}_1 to the line segment $\mathbf{s}_j^{(\text{opt})} \mathbf{s}_{j+1}^{(\text{opt})}$ is expressed as:

$$d_{\mathbf{v}_1} = r \cdot \sin(\frac{\pi}{3} + \omega - \gamma) - r \cdot \sin \omega \,. \tag{A.1}$$

To minimize $d_{\mathbf{v}_1}$, $\gamma = \arg \min_{\gamma} d_{\mathbf{v}_1} = \max \gamma = \theta$. Then get the derivative $\frac{d}{d\omega} d_{\mathbf{v}_1}$ to

find ω such that $d_{\mathbf{v}_1}$ is minimized, where $\frac{d}{d\omega}d_{\mathbf{v}_1} = r \cdot \cos(\frac{\pi}{3} + \omega - \theta) - r \cdot \cos \omega$. Since

 $\theta \le \frac{\pi}{4} < \frac{\pi}{3}$, it is obtained that $\frac{d}{d\omega} d_{\mathbf{v}_1} < 0$. Thus $\omega = \arg\min_{\omega} d_{\mathbf{v}_1} = \max \omega = \omega_{\max}$. The

minimum $d_{\mathbf{v}_1}$ is derived as:

$$\min d_{\mathbf{v}_1} = r \cdot \sin(\frac{\pi}{3} + \omega_{\max} - \theta) - r \cdot \sin \omega_{\max} \,. \tag{A.2}$$

The distance from $d_{\mathbf{v}_2}$ to the line segment is expressed as:

$$d_{\mathbf{v}_2} = r \cdot \sin(\frac{\pi}{3} + \gamma - \omega) + r \cdot \sin \omega \,. \tag{A.3}$$

To minimize $d_{\mathbf{v}_2}$, $\gamma = \arg \min_{\gamma} d_{\mathbf{v}_2} = \min \gamma = -\theta$. Then get the derivative $\frac{d}{d\omega} d_{\mathbf{v}_2}$ to find

 ω such that $d_{\mathbf{v}_2}$ is minimized, where $\frac{d}{d\omega}d_{\mathbf{v}_2} = -r \cdot \cos(\frac{\pi}{3} - \theta - \omega) + r \cdot \cos \omega$.

The minimum $d_{\mathbf{v}_{2}}$ can be obtained when:

$$\omega = \begin{cases} \omega_1 = \omega_{\min} = 0 \\ \omega_2 = \omega_{\max} = \arcsin(2\sin\frac{\theta}{2}) \\ \omega_3 = \arg(\frac{d}{d\omega}d_{v_2} = 0) = \frac{\pi}{6} - \frac{\theta}{2} \end{cases}$$
(A.4)

Thus, the minimum $d_{\mathbf{v}_2}$ can be obtained as:

$$d_{\mathbf{v}_{2}} = \begin{cases} d_{\mathbf{v}_{2}}^{(1)} = r \cdot \sin(\frac{\pi}{3} - \theta), \omega_{1} = 0 \\ d_{\mathbf{v}_{2}}^{(2)} = r \cdot \sin(\frac{\pi}{3} - \theta - \omega_{\max}) + r \cdot \sin \omega_{\max}, \omega_{2} = \omega_{\max} \\ d_{\mathbf{v}_{2}}^{(3)} = 2 \cdot r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}), \omega_{3} = \frac{\pi}{6} - \frac{\theta}{2} \end{cases}$$
(A.5)

Notice that $d_{\mathbf{v}_2}^{(1)} = r \cdot \sin(\frac{\pi}{3} - \theta) = 2 \cdot r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}) \cdot \cos(\frac{\pi}{6} - \frac{\theta}{2}) < d_{\mathbf{v}_2}^{(3)}$ and $d_{\mathbf{v}_2}^{(2)} = r \cdot d_{\mathbf{v}_2}^{(3)}$

$$\sin(\frac{\pi}{3} - \theta - \omega_{\max}) + r \cdot \sin \omega_{\max} = 2 \cdot r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}) \cdot \cos(\frac{\pi}{6} - \frac{\theta}{2} - \omega_{\max}) \le d_{\mathbf{v}_2}^{(3)}.$$
 However, the

relationship between $d_{\mathbf{v}_2}^{(1)}$ and $d_{\mathbf{v}_2}^{(2)}$ is all possible since $\cos(\frac{\pi}{6} - \frac{\theta}{2} - \omega_{\text{max}})$ can be bigger

than, smaller than, or equal to $\cos(\frac{\pi}{6} - \frac{\theta}{2})$.

In summary,
$$\min d_{\mathbf{v}_1} = r \cdot \sin(\frac{\pi}{3} + \omega_{\max} - \theta) - r \cdot \sin \omega_{\max}$$
; $\min d_{\mathbf{v}_2} = \min\{d_{\mathbf{v}_2}^{(1)}, d_{\mathbf{v}_2}^{(1)}\}$

$$d_{\mathbf{v}_2}^{(2)}$$
 . Note that $\min d_{\mathbf{v}_1} = r \cdot \sin(\frac{\pi}{3} + \omega_{\max} - \theta) - r \cdot \sin \omega_{\max} = 2r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}) \cdot \cos(\frac{\pi}{6} - \frac{\theta}{2})$

$$-\frac{\theta}{2} + \omega_{\max}), \ d_{\mathbf{v}_2}^{(1)} = r \cdot \sin(\frac{\pi}{3} - \theta) = 2r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}) \cos(\frac{\pi}{6} - \frac{\theta}{2}), \ \text{and} \ d_{\mathbf{v}_2}^{(2)} = r \cdot \sin(\frac{\pi}{3} - \theta)$$

 $-\omega_{\max}) + r \cdot \sin \omega_{\max} = 2r \cdot \sin(\frac{\pi}{6} - \frac{\theta}{2}) \cos(\frac{\pi}{6} - \frac{\theta}{2} - \omega_{\max}).$ Therefore, it can be obtained that $\min d_{\mathbf{v}_1} | \{\gamma = \theta, \omega_{\max}\} \le \min d_{\mathbf{v}_2} | \{\gamma = -\theta, \omega_1 = 0\} \text{ and } \min d_{\mathbf{v}_1} | \{\gamma = \theta, \omega_{\max}\} \le \min d_{\mathbf{v}_2} | \{\gamma = -\theta, \omega_2 = \omega_{\max}\}.$

Appendix B

B.1 Proof of the Factor Sensitivity

Consider a failed test $m_{ij} \notin [g_{1j}, g_{2j}]$, and denote the sum of excursions of the other tests

by,
$$\sum Excursion = \sum_{\mathcal{F}=1}^{\mathcal{F}} \sum_{\mathcal{L}=1}^{\ell} Excursion_{ij}$$
, $\mathcal{F} \neq i$, $\mathcal{L} \neq j$.

The sensitivity of the Amplitude Factor is derived as follows:

$$S_{3} = \frac{\sum Excursion + Excursion_{ij}}{\sum Excursion + Excursion_{ij} + f \cdot \ell} \cdot 100 - \frac{\sum Excursion}{\sum Excursion + f \cdot \ell} \cdot 100$$

$$= \frac{Excursion_{ij} \cdot f \cdot \ell}{(\sum Excursion + Excursion_{ij} + f \cdot \ell)(\sum Excursion + f \cdot \ell)} \cdot 100$$

$$= \frac{Excursion_{ij} \cdot f \cdot \ell}{(\sum Excursion + f \cdot \ell) \cdot Excursion_{ij} + (\sum Excursion + f \cdot \ell)^{2}} \cdot 100$$

$$= \frac{f \cdot \ell}{(\sum Excursion + f \cdot \ell) + \frac{(\sum Excursion + f \cdot \ell)^{2}}{Excursion_{ij}}} \cdot 100$$