An Integrated Online Path Planning and Control Approach for Robotic Sensor Networks

by

Guoxian Zhang

Department of Mechanical Engineering and Materials Science

Date: _____

Approved:

_____
Silvia Ferrari, Supervisor

_____
David L Banks

_____
Devendra P Garg

_____
Pankaj K Agarwal

_____
Xiaobai Sun

Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mechanical Engineering and Materials Science
in the Graduate School of Duke University
2010

Abstract
(Warlockery)

An Integrated Online Path Planning and Control Approach for Robotic Sensor Networks

by

Guoxian Zhang

Department of Mechanical Engineering and Materials Science

Date: _____

Approved:

_____
Silvia Ferrari, Supervisor

_____
David L Banks

_____
Devendra P Garg

_____
Pankaj K Agarwal

_____
Xiaobai Sun

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Mechanical Engineering and Materials Science in the Graduate School of Duke University 2010

Abstract

This dissertation addresses an information-driven sensor path planning problem which has various applications such as robot cleaning, environment monitoring, and manufacturing. Information-driven sensor path planning is concerned with planning the measurements of a sensor or a sensor network in order to support sensing objectives, such as target detection, classification and localization, based on prior information. When the sensor's field-of-view or visibility region is bounded, the sensor's position and orientation determine what targets can be measured at any given time. Therefore, the sensor path must be planned in concert with the measurement sequence. When sensors are installed on robotic platforms and are deployed in an obstacle-populated environment, the sensor path must also avoid collisions between the platform and the obstacles or other robotic sensors. Addressing this sensor path planning problem, this dissertation first presents a general and systematic approach for deriving information value functions that represent the expected utility of sensor decisions in a canonical sensor planning problem. The resulting information functions and search strategies are compared through extensive numerical simulations involving direct-search, alert-confirm, task-driven, and log-likelihood-ratio search strategies, and the maximum *a-posteriori*, maximum-likelihood, and Neyman-Pearson decision rules. After that a novel off-line information roadmap method is developed to navigate single robotic sensor in which obstacles, targets, sensor's platform and field of view are represented as closed and bounded subsets of an Euclidean workspace.

The information roadmap is sampled from a normalized information theoretic metric that favors samples with a high value of information in configuration space. Finally, when multiple robotic sensors are deployed in the workspace, and information of the workspace such as geometry, location, and prior measurements on targets and obstacles can become available online, another novel sensor path planning method, named information potential method, is proposed to take into account the new information obtained over time. Targets with high information value tend to have high probability to be measured by the robotic sensor network. A hybrid control system is utilized to coordinate and control each robotic sensor in the network to detect and measure obstacles and targets in the workspace. The potential function is also utilized to generate the milestones in a local probabilistic roadmap method to help robotic sensors escape their local minima.

The proposed methods are applied to a landmine classification problem to plan the path of a robotic sensor network in which each robot is equipped with a ground-penetrating radar. Other sensors, such as infrared sensors on unmanned aerial vehicles (UAVs), are utilized *a priori* for target detection and cursory classification. In the off-line sensor path planning applications for a single robotic sensor, experiments show that paths obtained from the information roadmap exhibit a classification efficiency significantly higher than that of existing robot motion strategies. Also, the information roadmap can be used to deploy non-overpass capable robots that must avoid targets as well as obstacles. Then in the multiple online robotic sensor network path planing applications, experiments show that path obtained from the information potential method takes advantages of the online information and coordination among robotic sensors, and the results show that the information potential method outperforms other strategies such as rapidly-exploring random trees and classical potential field methods that does not take target information value into account.

To my parents, Zuolin and Fengchang, my wife, Jiali, and my son Andrew.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

Symbols

A	Set of sensor platforms $A = \{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_r\}$
$\mathcal{A}$	Geometry of sensor platform
$\mathcal{A}_i$	Geometry of ith sensor platform
B	Set of obstacles $B = \{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_n\}$
$\mathcal{B}_i$	Geometry of ith obstacle, $i = 1, 2, \ldots, n$
$\mathcal{C}$	Configuration space
$\mathcal{C}_{free}$	Free configuration space
$\mathcal{CB}$	C-obstacle region
$\mathcal{CT}$	C-target region
c	Total number of cells in $\mathcal{K}$
c_e	Total number of empty cells in $\mathcal{K}$
c_r	Total number of cells in $\mathcal{K}$ with state value x^r
c_t	Total number of target in $\mathcal{K}$
$D(p\|\|q)$	Kullback-Leibler divergence between p and q
$D_\alpha(p\|\|q)$	α-divergence between p and q
$D(\tau)$	Scaled Euclidian distance along path τ
$\mathcal{F}_\mathcal{A}$	Moving Cartesian frame embedded in $\mathcal{A}$
$\mathcal{F}_{\mathcal{A}_i}$	Moving Cartesian frame embedded in $\mathcal{A}_i$, $i = 1, 2, \ldots, r$
$\mathcal{F}_\mathcal{W}$	Cartesian frame embedded in $\mathcal{W}$

$F_c(k)$	Rate of correct classification at time k
$F_{fa}(k)$	Frequency of false alarm at time k
$F_r(k)$	Rate of correct classification for high-risk targets at time k
$H(X)$	Shannon entropy of X
$H_{R_\alpha}(X)$	Rény's entropy of order α
$I(X;Y)$	Mutual information between X and Y
I_A	Index set of A
I_B	Index set of B
I_T	Index set of T
$\mathcal{M}^k$	Set of measurements up to time k
M_i	Set of measurements on $\mathcal{T}_i$
$M(\tau)$	Set of measurements along path τ
$N(j)$	The index of targets assigned to the jth robotic sensor
n	Number of cells with a state value estimated correctly
n_{fa}	Number of cells that are empty but declared to contain a target
n_r	Number of cells with a high risk state value that is estimated correctly
$p(x)$	Probability density function or Probability mass function for random variable X
$q(x)$	Probability density function or Probability mass function for random variable X
$\mathbf{q}$	Robot configuration
$\mathbf{q}_0$	Robot initial configuration
$\mathbf{q}_i$	The ith robot configuration
T	Set of targets $T = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m\}$
$\mathcal{T}_i$	Geometry of ith target, $i = 1, 2, \ldots, m$
S	Set of sensor FOV, $S = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_r\}$

$\mathcal{S}$	Geometry of sensor FOV
$\mathcal{S}_i$	Geometry of the ith sensor FOV, $i = 1, 2, \ldots, r$
$U(\mathbf{q})$	Potential function
$U^j(\mathbf{q})$	Potential function for the jth robotic sensor
$U_{att}(\mathbf{q})$	Attractive potential function
$U^j_{att}(\mathbf{q})$	Attractive potential function for the jth robotic sensor
$U^j_r(\mathbf{q})$	Repulsive potential function for the jth robotic sensor by other robotic sensors
$U_{rep}(\mathbf{q})$	Repulsive potential function
$V(X)$	Information potential of X
V_i	Information value ith garget
$\mathcal{W}$	Geometry of workspace
X_i	Random variable for ith cell state, $i = 1, \ldots, c$
X^k or $\mathbf{X}^k$	Random variable for cell state at time step k
x^r	High risk target state value
y_i	State of the ith target in IRM
Z	Sensor measurement
$\mathcal{Z}$	Sample space of Z
Z_i or $\mathbf{Z}_i$	Measurement at the ith cell or target, $i = 1, \ldots, c$
Z^k	Measurement at time step k
z^k	Observation of measurement at time step k
θ	Parameter of sensor model distribution, or robotic sensor heading angle
κ_i	ith cell, $i = 1, 2, \ldots, c$
λ^k	Random variable for environmental conditions at time step k.
Λ	Sample space of λ^k
μ	The parameter matrix in the growth curve model

μ_{ij}	The element at ith row jth column of matrix μ
$\rho_{goal}(\mathbf{q})$	The distance between $\mathbf{q}$ and the goal
$\rho_i^b(\mathbf{q})$	The distance between $\mathbf{q}$ and the ith obstacle
$\rho_i^t(\mathbf{q})$	The distance between $\mathbf{q}$ and the ith target
τ	Robotic sensor path
τ_i	The ith robotic sensor path
τ^*	Optimal robotic sensor path
$\hat{\varphi}_C$	Expected CS information function
$\hat{\varphi}_D$	Expected KL divergence
$\hat{\varphi}_{D_\alpha}$	Expected α divergence
$\hat{\varphi}_I$	Conditional mutual information
$\hat{\varphi}_{R_2}$	Expected quadratic entropy reduction
$\hat{\varphi}_V$	Expected information potential gain

Abbreviations

AS	Alert-confirm search
APM	Anti-personnel mines
ATM	Anti-tank mines
BN	Bayesian network
CL	Level of confidence
CLUT	Clutter objects
CPT	Conditional probability table
CS	Cauchy-Schwartz
DS	Direct search
EDG	Expected discrimination gain
EER	Expected entropy reduction (conditional mutual information)

EIPG	Expected information potential gain
EQER	Expected quadratic entropy reduction
FOV	Field-of-view
GPR	Ground-penetrating radar mounted
IPM	Information potential method
IRM	Information roadmap method
IS	Information-driven search
KL	Kullback-Leibler
LLR	Log-likelihood-ratio
MTI	Moving target indicator
PDF	Probability density function
PMF	Probability mass function
PRMs	Probabilistic roadmap methods
SNR	Signal-to-noise ration
TS	Task-driven search
UAV	Unmanned air vehicle
UXO	Unexploded ordnance

Acknowledgements

I would first like to thank my advisor, Dr. Silvia Ferrari. Your help, financial support, and dedication to me over the course of my graduate studies were invaluable. Your consistent motivation kept me focused every day, and your academic adventure enhances my interests and passion in research.

I would also like to give special mention to those faculty who have served on my Ph.D. qualification, preliminary exam, and final defense committees for their time and effort: Dr. Devendra Garg, Dr. David Banks, Dr. Xiaobai Sun, and Dr. Pankaj Agarwal. In addition, I would like to thank those who have contributed to my better understanding of this research: Dr. Rafael Fierro and Dr. Tom Wettergren. A special thank you to my great labmates and fellow graduate students, who were always available and willing to help along the way: Kelli Baumgartner, Chenghui Cai, Gianluca Di Muro, Andrew Tremblay, Wenjie Lu, Greg Foderaro, Greyson Daugherty, Brian Bernard, Ashleigh Swingler, Keith Rudd, Vikram Raju, and Hersh Tapadia.

I wish to dedicate this dissertation to my family. To my parents, Zuolin and Fengchang. Thanks for having provided me a fantastic environment since I was a kid. Your encouragement to my ambitions gave me the confidence that I can do anything. To my better half, Jiali. This will never happen without your support, understanding, and love. To my little buddy Andrew. Just a smile from you provides me all the power to overcome every difficulty in front of me. Your mommy and I will hold your hands and show you the amazing world.

Introduction

This dissertation addresses the problem of sensor path planning for a robotic sensor network. Sensor path planning is concerned with planning the measurements of a sensor or a sensor network in order to support sensing objectives, such as target detection, classification and localization. When the sensor's field-of-view (FOV) or visibility region is bounded, the sensor's position and orientation determine what targets can be measured at any given time. Therefore, the sensor path must be planned in concert with the measurement sequence. When sensors are installed on robotic platforms and are deployed in an obstacle-populated environment, the sensor path must also avoid collisions between the platform and the obstacles or other robots [2–4]. The problem of planning the sensor path in order to maximize the information profit, while minimizing the distance traveled and avoiding collisions with the obstacles, is referred to as treasure hunt [5–7]. This problem is relevant to many sensor applications such as, robotic mine hunting [8], cleaning [2], and robotic games [9], as well as the monitoring of urban environments [10], underwater objects [11], manufacturing plants [12], and endangered species [13].

In sensor path planning, prior information such as sensor models, environmental conditions, and prior measurements may be available. In many applications, such as monitoring, maintenance, and surveillance operations, the set of all measurements that can be acquired by a sensor significantly exceeds its available time and processing capabilities [12]. The sensor can then be viewed as an information-gathering

agent that must make decisions on the mode and measurement sequence, in order to optimize the performance of a sensing task, such as detection, classification, and tracking [14–16]. A basic difficulty in sensor planning consists of assessing the sensor performance that will result from the sensor decisions prior to obtaining the sensor measurements [6, 17, 18]. Many sensing tasks can be reduced to the problem of estimating one or more random variables from partial or imperfect measurements [19]. Therefore, the sensor performance depends on the amount of information, or lack thereof, associated with these variables, and the utility of the sensor decisions can be represented by the expected information value.

Several information theoretic functions have been proposed in the literature to measure the information value in sensor planning and management problems. Relative entropy was used in [20] to solve a multisensor-multitarget assignment problem, and in [21, 22] to manage agile sensors with Gaussian models for target detection and classification. Shannon entropy and the Mahalanobis distance were applied and compared in [18] for sensor selection in *ad-hoc* sensor networks. Shannon entropy was also used in [23,24] for tracking a moving target using a Kalman filter. Recently, mutual information has been used for sensor planning in multi-target detection, classification, and feature inference by ground-penetrating radars and infrared sensors in [6, 25]. An approach based on mutual information was also presented in [26] for adjusting the parameters of a camera in an object recognition application.

Due to the significant differences between the aforementioned sensing tasks and applications, there presently exists no systematic and general approach for estimating information theoretic functions prior to obtaining the sensor measurements. Furthermore, little work has been done on comparing these information functions on canonical sensor problems, and on generalizing the results beyond a particular sensor type and application. The comparative study closest to the one presented in this dissertation pertains the optimal choice of the α parameter in the Rènyi divergence [22]. It

was determined that the optimal value of α for a multitarget tracking application involving a simulated moving target indicator (MTI) sensor is 0.5. An empirical study showing that task-driven approaches may slightly outperform information-driven approaches was recently conducted in [27] using a synthetic-aperture-radar model with two modes of operation, and eight possible target locations or cells. However, the study in [27] only considered one sensor model and one information function, i.e. the Rènyi divergence with $\alpha = 0.5$.

In this dissertation, a general approach is presented for estimating and comparing six information value functions derived from mutual information, Rènyi divergence, Kullback-Leibler (KL) divergence, information potential, quadratic entropy, and the Cauchy-Schwartz distance. Although the Cauchy-Schwartz distance and information potential have not been previously applied to sensor planning, they are considered here because they have been successfully implemented for blind-source separation, feature extraction, and machine learning [28–30]. As reviewed in [31], sensor performance depends on the search strategy, and on the decision rule used for hypothesis acceptance and rejection. In the information-driven approach, the search strategy consists of selecting the sensor mode and measurement sequence that maximize the information value function. Other effective search strategies that have been proposed in the literature and are implemented in this dissertation are direct search, alert-confirm search, task-driven or Bayes' risk search, and log-likelihood-ratio search [27, 32, 33]. Therefore, in this dissertation, these search strategies also are evaluated and compared to the information-driven approach, using the maximum likelihood, and maximum *a-posteriori* decision rules, and five sensor models obtained from Bernoulli, Poisson, binomial, and mixture-of-binomials distributions, and a noisy power law. All approaches are implemented on a canonical problem formulation that is relevant to a variety of applications [1,22,34]. The effects of prior information and data fusion are investigated by simulating prior distributions with

three levels of information-to-noise ratio.

With various information metric analyzed in this dissertation, the prior information such as sensor models, environmental conditions, and prior measurements are able to be utilized to compute the information value of measuring a target in optimal search strategies [35,36] for sensor path planning, and results in a better performance than other existing approaches such as coverage path-planning [3, 37], random [3], and grid [35] search strategies. One of the popular path planing algorithm, cell decomposition [5, 6], has been successfully developed for solving geometric sensor path planning problems, such as the treasure hunt, when a prior model of the obstacles and the targets is available. Although it has the advantage of being resolution complete, this method is computationally intensive and may not be applicable to problems with many robotic sensors, obstacles, and targets. Also, cell decomposition is not easily applicable when information about the obstacles and targets is obtained incrementally over time. This dissertation presents two new sensor path planning methods, referred to as information roadmap method and information potential method, that compute sensor paths for geometric sensing.

In the information roadmap method (IRM), the probabilistic roadmap approach or (PRMs) is combined with information-driven sensor planning to generate the path for a robotic sensor. This off-line method takes into account the robotic sensor's platform and field of view, as well as the geometry and position of multiple fixed targets and obstacles in the workspace. Traditionally, PRMs have been used to plan the motions of a robot with geometry $\mathcal{A}$, in order to avoid collisions with multiple fixed obstacles in a workspace $\mathcal{W}$ [38–41]. While robot path planning typically aims to optimize a deterministic additive function such as Eucledian distance, sensor planning aims to optimize a stochastic sensing objective that is not necessarily additive. Another basic difficulty in sensor planning is that, although the measurements ultimately determine the sensor performance, they cannot be factored into the planning

problem because the sensor's position must be planned prior to obtaining the sensor's measurements [18, 20–22, 42]. Recently, several authors have shown that this difficulty can be overcome by an approach known as information-driven sensor planning, which uses information theoretic metrics to estimate the measurements' value prior to deploying the sensor [18, 20–22].

The information roadmap method developed in this dissertation combines the advantages of PRMs and information-driven sensor planning, by sampling a normalized information theoretic metric defined over the robot free configuration space. The information theoretic metric can be various information functions of potential targets, as introduced in Chapter 3. A new hybrid sampling strategy is presented in Chapter 4 to generate an information roadmap that contains a high density of milestones with high information metric value, as well as milestones that capture the connectivity of the free configuration space. A new query phase is also presented for searching the information roadmap using an A^*-type algorithm that returns the path of maximum information profit whenever one exists, and returns failure otherwise.

The information roadmap method is demonstrated by planning the path of a mine hunting ground-penetrating radar (GPR) installed on a ground robot. The results show that the method outperforms existing sensor path-planning methods that are applicable to geometric sensing (namely, complete coverage [3, 37] and random search [3]) under a wide range of workspace conditions and geometries, increasing the average classification efficiency by up to one order of magnitude. Also, the information roadmap can be used to deploy non-overpass capable robotic sensors that may be damaged or destroyed when driving over landmines. As a result, the robotic sensor avoids collisions with potential mines, while obtaining measurements from them, both in wide-open regions of the workspace and inside narrow passages.

The second method, referred to as information potential method (IPM), is developed for online sensor path planing such that cursory measurements of targets

and obstacles obtained incrementally over time are utilized for planning sensor path. Although several potential fields have been developed for robot motion planning, existing techniques are not directly applicable to geometric sensor path planning because they do not take into account the geometries of the targets and of the sensor FOV, and do not consider the information value of the sensor measurements [43–49]. In the classical potential field method [50–53], for example, the robot's objective is to navigate the workspace to reach a goal configuration by following the negative gradient of a potential function that is designed to provide a repulsive potential near the obstacles and an attractive potential toward the goal configuration. Although the potential field method is well suited to on-line motion planning and to convergence analysis, its effectiveness is limited by the tendency of the robot to get stuck in local minima of the potential function [54]. An effective approach for escaping local minima is to follow a new local path generated through a random-walk algorithm. Another way to solve this problem is to construct a potential field without any local minima besides the goal. Functions with this probability are called Morse function or Harmonic function [55–58]. The disadvantage of using potential fields free of undesired local minima is that their computational cost is usually high. Especially in on-line path planning, a potential over the whole configuration space generated by obstacle or target needs to be recomputed every time when a new target or obstacle is detected, and the computation is usually expensive. Potential field has been previously combined with PRMs. In Sensor-based robot path planning, Kazemi [47] uses the potential field approach to bias the distribution of random nodes. In [59] the potential energy of a protein is utilized to sample the milestones of the PRMs. These methods, however, does not applicable for sensor path planning since no sensing objective is considered.

In this dissertation, a novel potential function is presented for generating attractive potentials toward the targets, based on their geometries, locations and informa-

tion from prior measurements. The potential function is also utilized to construct a local probabilistic information roadmap with which the robotic sensor can escape from the local minimum.

One of the key results presented in this dissertation is that the same adaptive potential function can be used to generate the navigation potential and the roadmap, thereby providing a natural framework for integrating the two approaches. A method is presented for using the adaptive information roadmap to escape local minima of the potential function in place of random-walk algorithms that can lead the sensor to regions of poor information value. In a cooperative sensor network, the adaptive information of targets and obstacles can also be exchanged between robots and used to plan the path of other robots based on the latest sensed information. By defining a potential function in configuration space, this approach accounts for the geometries and positions of the obstacles, targets, sensor's platform and FOV. The information value of the targets is represented by defining the potential function in terms of their information value conditioned on prior sensor measurements. As a result, the sensors visit targets that offer the best tradeoff between distance and information value, and through this on-line sensor path planning approach, they adapt their paths based on new sensor measurements obtained from targets or obstacles that were previously undetected.

This dissertation is organized as follows. The sensor path planning problem and assumptions addressed in this dissertation are presented in Chapter 1. Background on information functions and robot motion planning is reviewed in Chapter 2. A general and systematic approach for estimating the information functions prior to obtaining the sensor measurements is presented in Chapter 3. The information roadmap method is developed and demonstrated for a single robotic sensor path planning in Chapter 4. Chapter 5 presents the information potential method for planning the paths of multiple robotic sensors that obtain, and share via wireless communications,

information about the targets and obstacles incrementally over time.

1

Problem Formulation and Assumptions

This dissertation addresses the problem of planning the path of a robotic sensor, or robotic sensor network, where each robotic sensor has platform geometry $\mathcal{A}$, and a field-of-view geometry $\mathcal{S}$, and navigates a workspace $\mathcal{W}$ for the purpose of classifying multiple fixed targets based on new (posterior) and prior sensor measurements, and environmental information. $\mathcal{W}$ is assumed to be a compact and bounded subset of a two- or three-dimensional Euclidian space. Both $\mathcal{A}$ and $\mathcal{S}$ are assumed to be rigid convex sets, and $\mathcal{S}$ has a fixed position and orientation with respect to $\mathcal{A}$. Prior measurements may be available from airborne sensors, environmental maps, or from another type of sensor installed on the robotic sensor plaftorm, and are used to estimate the geometry and location of targets and obstacles in $\mathcal{W}$ [60,61]. The robotic sensor workspace $\mathcal{W}$ is populated with n fixed obstacles $B = \{\mathcal{B}_1, \ldots, \mathcal{B}_n\} \subset \mathcal{W}$, and m fixed targets $T = \{\mathcal{T}_1, \ldots, \mathcal{T}_m\} \subset \mathcal{W}$, with $\mathcal{B}_i \cap \mathcal{T}_j = \emptyset$ for $\forall i \in I_B$ and $\forall j \in I_T$, where I_B and I_T are the index sets of B and T.

Let $\mathcal{F}_\mathcal{A}$ be a moving Cartesian frame embedded in $\mathcal{A}$. Then, every point of $\mathcal{A}$ and every point of $\mathcal{S}$ have a fixed position with respect to $\mathcal{F}_\mathcal{A}$, and a configuration vector $\mathbf{q} = [x\ y\ \theta]^T \in SE(2)$ can be used to specify the position (x, y) and orientation θ of

both $\mathcal{A}$ and $\mathcal{S}$ with respect to a fixed inertial frame $\mathcal{F}_{\mathcal{W}}$, embedded in $\mathcal{W}$. Obstacles and targets are also assumed to be fixed and rigid in $\mathcal{W}$, such that every point of $\mathcal{B}_i$, for $\forall i \in I_B$, and every point of $\mathcal{T}_j$, $\forall j \in I_T$, has a fixed position with respect to $\mathcal{F}_{\mathcal{W}}$. Let the configuration space $\mathcal{C}$ denote the space of all possible robot configurations. A C-obstacle is a subset of $\mathcal{C}$ that causes collisions with at least one obstacle in B, i.e., $\mathcal{CB}_i \equiv \{\mathbf{q} \in \mathcal{C} \mid \mathcal{A}(\mathbf{q}) \cap \mathcal{B}_i \neq \emptyset\}$, where $\mathcal{A}(\mathbf{q})$ denotes the subset of $\mathcal{W}$ occupied by the platform geometry $\mathcal{A}$ when the robot is in the configuration $\mathbf{q}$. The union of all C-obstacles obtained from B is referred to as the C-obstacle region. Thus, in searching for targets in $\mathcal{W}$, the robotic sensor is free to rotate and translate in the free configuration space, which is defined as the complement of the C-obstacle region $\mathcal{CB}$ in $\mathcal{C}$, i.e., $\mathcal{C}_{free} = \mathcal{C} \backslash \mathcal{CB}$ [50]. The path of the robotic platform's centroid is defined as a continuous map $\tau : [0, 1] \rightarrow \mathcal{C}$, with $\mathbf{q}_0 = \tau(0)$ and $\mathbf{q}_f = \tau(1)$, where $\mathbf{q}_0$ and $\mathbf{q}_f$ are the initial and goal configurations, respectively. Since $\mathcal{S}$ is mounted on $\mathcal{A}$, the path τ determines the targets in $\mathcal{W}$ that can be measured by the robotic sensor, while traveling from $\mathbf{q}_0$ to $\mathbf{q}_f$.

In standard estimation theory, a sensor that obtains a vector of measurements $\mathbf{Z} \in \mathcal{Z} \subset \mathbb{R}^r$ in order to estimate an unknown state vector $\mathbf{X} \in \mathcal{X} \subset \mathbb{R}^n$ is modeled as,

$$\mathbf{Z}^k = \mathbf{h}(\mathbf{X}^k, \boldsymbol{\lambda}^k) \tag{1.1}$$

where $\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^{\wp} \rightarrow \mathbb{R}^r$ is a deterministic vector function that is possibly nonlinear, $\boldsymbol{\lambda} \in \mathbb{R}^{\wp}$ is the random vector representing the sensor characteristics, such as sensor mode, environmental conditions, and sensor noise or measurement errors. It is assumed that the sensor model is time invariant and k is the discrete time index. In many sensor applications, however, the state, the measurements, and the sensor characteristics also are random vectors. Therefore, a more general observation or measurement model that has been adopted in the literature is the joint probability

mass function (PMF) $p(\mathbf{Z}^k, \mathbf{X}^k, \boldsymbol{\lambda}^k)$. This joint PMF may be given by the factorization,

$$p(\mathbf{Z}^k, \mathbf{X}^k, \boldsymbol{\lambda}^k) = p(\mathbf{Z}^k \mid \mathbf{X}^k, \boldsymbol{\lambda}^k) p(\mathbf{X}^k) p(\boldsymbol{\lambda}^k) \tag{1.2}$$

which includes the conditional PMF $p(\mathbf{Z}^k \mid \mathbf{X}^k, \boldsymbol{\lambda}^k)$, the priors $p(\mathbf{X}^k)$ and $p(\boldsymbol{\lambda}^k)$, and assumes that $\mathbf{X}^k$ and $\boldsymbol{\lambda}^k$ are independent. Various sensors, including infrared, ground penetrating radars, and synthetic aperture radars have been modeled as (1.2) in target detection, classification, and tracking applications [1, 21, 22, 27, 34]. In this dissertation, the joint PMF in (1.2) is considered to be the sensor model, and is assumed known. Since the sensor model holds for all targets and environmental conditions, and can be assumed to remain constant over time, (1.2) can be written as $p(\mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\lambda}_i)$ for every target $\mathcal{T}_i \in T$.

The purpose for deploying the robotic sensor in $\mathcal{W}$ is to obtain measurements from a subset of targets in T. To each target $\mathcal{T}_i \in T$, there is associated an information value denoted by V_i that is computed by the information function defined in Chapter 3, and represents the expected benefit of making measurements from $\mathcal{T}_i$, based on the sensor model and on prior information. V_i can be considered as the expected uncertainty reduction for target features or classification.

While the platform $\mathcal{A}$ must avoid collisions with the obstacles B and other robotic sensors, the sensor's FOV $\mathcal{S}$ must intersect $\mathcal{T}_i$ in order to obtain the measurements $\mathbf{Z}_i$. Since $\mathcal{S}$ is mounted on $\mathcal{A}$, the platform motion must be planned in concert with the sensor measurements, and the path τ must simultaneously avoid obstacles and other robotic sensors while searching for targets. Let the measurement set of a robotic sensor along a path τ be defined as $Z(\tau) = \{\mathbf{Z}_i \mid \mathcal{T}_i \cap \mathcal{S}(\mathbf{q}) \neq \emptyset,\ \tau(s) = \mathbf{q},\ s \in [0,1],\ i \in I_T\}$, where $\mathcal{S}(\mathbf{q})$ is the subset of $\mathcal{W}$ occupied by $\mathcal{S}$ at a configuration $\mathbf{q}$, along τ. Then, the robotic sensor path τ between $\mathbf{q}_0$ and $\mathbf{q}_f$ must avoid all obstacles and other robotic sensors in $\mathcal{W}$, and maximize the information value of the

measurement set $Z(\tau)$ by traveling the minimum distance. These above formulation and assumptions apply to the remaining chapters; however in specific problems, some extra descriptions and assumption may apply.

2

Background

2.1 Information Theoretic Functions

Information theoretic functions have been used to evaluate the information value of sensor measurements in a wide range of applications. As pointed out in [18], a natural choice for measuring information value is entropy. Shannon entropy measures the uncertainty of a discrete random variable X, with finite range $\mathcal{X}$, from its PMF $p_X(x) = \Pr(\{X = x\})$ for $x \in \mathcal{X}$, and is defined as

$$H(X) = -\sum_{x \in \mathcal{X}} p_X(x) \log_2 p_X(x) \tag{2.1}$$

Since the computation of (2.1) requires knowledge of the PMF, it cannot be used to compute the information value, because the posterior PMF of X, or belief state, is unknown before the measurements are obtained [18]. Furthermore, the optimization of entropy-based functions is usually ill posed, because entropy is nonadditive, and myopic, i.e., it does not consider the effects of prior measurements on those that are performed subsequently [14, 42]. Another shortcoming of (2.1) is that it is not a true metric, because it is nonsymmetric, and it does not satisfy the triangle inequality

[6, 42]. In the remainder of this thesis, the PMF $p_X(x)$ will be represented by the shorthand notation $p(x)$. Also, uppercase characters are used to denote discrete random variables, and lower case characters are used to denote real numbers, such as the numerical values of random variables.

The Rény information or α-divergence has been proposed in [62] as a means for quantifying the change in the belief state brought about by the sensor measurements. It is based on the Rény's entropy of order α, defined as,

$$H_{R_\alpha}(X) = \frac{1}{\alpha - 1} \log_2 \sum_{x \in \mathcal{X}} p^\alpha(x) \tag{2.2}$$

which relates to (2.1) through the properties $\lim_{\alpha \to 1} H_{R_\alpha}(X) = H(X)$, and $H_{R_\alpha}(X) \geq H(X) \geq H_{R_\beta}(X)$ if $1 > \alpha > 0$ and $\beta > 1$. Let the current belief state be represented by a PMF $q(x)$, and suppose a posterior distribution $p(x)$ is expected as a result of a sensor decision pertaining the sensor mode and measurement sequence. Then, the α-divergence,

$$D_\alpha(p \parallel q) = \frac{1}{\alpha - 1} \log_2 \sum_{x \in \mathcal{X}} p^\alpha(x)\, q^{1-\alpha}(x) \tag{2.3}$$

can be viewed as a measure of the difference between the two PMFs $q(x)$ and $p(x)$, where the α parameter represents the emphasis placed on the degree of differentiation between the tails of the distributions. In [22], the value $\alpha = 0.5$ was found to be optimal for representing the information value in multitarget tracking applications in which the two PMFs $q(x)$ and $p(x)$ are close. In the limit of $\alpha \to 1$, (2.3) can be shown to reduce to the KL divergence or relative entropy, defined as,

$$D(p \parallel q) = \sum_{x \in \mathcal{X}} p(x) \log_2 \frac{p(x)}{q(x)} \tag{2.4}$$

which was first applied to sensor planning in the seminal work by Kastella [21]. Like entropy, however, the Rény information and KL divergence functions are nonadditive,

nonsymmetric, and do not satisfy the triangle inequality.

An additive, symmetric, and non-myopic function based on conditional mutual information was recently proposed in [5], and successfully applied to multitarget detection and classification in [6]. Mutual information is a measure of the information content of one random variable about another random variable [63]. The conditional mutual information of two random variables X and Z, given Y, is defined as,

$$\begin{aligned} I(X;Z \mid Y) &= H(X \mid Y) - H(X \mid Z,Y) \\ &= \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} \sum_{z\in\mathcal{Z}} p(x,y,z) \log_2 \frac{p(x,z \mid y)}{p(x \mid y)p(z \mid y)} \end{aligned} \tag{2.5}$$

and represents the reduction in uncertainty in X due to knowledge of Z, when Y is given. Where, $H(X \mid Y)$ denotes the conditional entropy of X given Y, and is defined in [63]. Although (2.5) requires knowledge of the posterior PMF and, thus, of the sensor measurements, this difficulty can be circumvented by using the expected conditional entropy, as shown in Chapter 3. Mutual information can also be shown to be a concave function of $p(x \mid y)$ for fixed $p(z \mid x, y)$, using the approach in [63].

Based on the Cauchy-Schwartz (CS) inequality, the information function defined as,

$$C(p,q) = \log_2 \frac{\sum_{x\in\mathcal{X}} p^2(x) \sum_{x\in\mathcal{X}} q^2(x)}{\left[\sum_{x\in\mathcal{X}} p(x)q(x)\right]^2} \tag{2.6}$$

was proposed as a measure of the difference between two PMFs $p(x)$ and $q(x)$ in [28, 29], and used therein for blind-source separation and feature extraction. The information function in (2.6) is based on quadratic entropy, which is obtained from (2.2) by letting $\alpha = 2$. Quadratic forms are particularly well suited to numerical optimization because they are characterized by high convergence rates and smooth gradient variations near the minimum, and, in the absence of constraints, they typically do not exhibit multiple stationary points. Therefore, in this thesis, quadratic

entropy and the information potential, defined as,

$$V(X) = \sum_{x \in \mathcal{X}} p^2(x) \tag{2.7}$$

are also considered for sensor planning. Optimizing the information potential in (2.7) is equivalent to optimizing the quadratic entropy, because $H_{R_2}(X) = -\log_2 V(x)$. However, (2.2) and (2.7) cannot be utilized for sensor planning because they require knowledge of the posterior PMF. Instead, as shown in Chapter 3, they are utilized to derive two new information functions, the expected quadratic entropy and the expected information potential.

Information functions can also be defined for continuous random variables, based on their probability density functions [63]. The discrete variable formulation is adopted throughout this dissertation because it leads to a more efficient numerical implementation. The information theoretic functions reviewed above are used to derive information value functions for the canonical sensor problem presented in Chapter 3.

2.2 Bayesian Network Approach to Sensor Modeling

A common approach for modeling the sensor measurement process is to utilize a joint PMF of the relevant variables, which may include target classification and features, sensor measurements and parameters (or mode), and environmental conditions. The joint PMF of a particular sensor may be obtained by means of estimation algorithms (e.g., [18, 22]), or by learning algorithms using wavelets or mixtures of Gaussians [35, 64]. In this dissertation, we adopt the method presented in [1, 34], in which the PMF is learned from data and represented by a Bayesian network (BN) model. The advantages of BN models are that they can easily deal with many variables, they are accompanied by very efficient learning and inference algorithms, and they

provide a convenient factorization of the joint PMF that can be used to simplify the computation of posterior PMFs required by information theoretic metrics.

A Bayesian network model is a pair comprised of a directed graph and of a set of conditional probability tables (CPTs) that together specify the multivariate joint PMF of a set of discrete random variables known as the *universe* [65]. Every random variable in the BN universe is assumed to have a finite range, and is represented by a node in the graph. Arcs between the nodes represent conditional probability relationships between the variables. As shown in [1, 34], the BN model of a sensor measurement process is obtained by defining the universe as the set of all variables that influence the sensor measurements, such as, the set of sensor-mode parameters S, the set of environmental conditions E, the set of measurements M, the set of actual target features F to be inferred from M, and the target classification variable y, i.e., $U \equiv \{S, E, M, F, y\}$. Then, the BN arcs and CPTs are determined from a database of prior sensor data, using BN batch learning algorithms [66]. The database consists of several cases in which all variables in U are sampled by obtaining sensor measurements from several known targets, under known operating and environmental conditions [1].

After the BN model is determined, it specifies the joint PMF underlying the sensor measurements in terms of the recursive factorization,

$$P(U_i) \equiv P(S_i, E_i, M_i, F_i, y_i) = \prod_{u_j \in U_i} P(u_j \mid pa(u_j)) \tag{2.8}$$

$$= P(M_i \mid S_i, E_i, F_i) P(F_i \mid y_i) P(y_i) P(S_i) P(E_i) \tag{2.9}$$

which holds for any target $\mathcal{T}_i$, i.e., $\forall i \in I_T$. Where, $pa(u_j)$ denotes the set of parents of a node $u_j \in U_i$, and factors in (2.8) are the BN CPTs. Since the parents of a node u_j are all the nodes in the BN with an outgoing arc to u_j, the factorization in (2.8) reflects the BN graph structure, which is learned from data and, thus, depends on the

sensor type. The factorization in (2.9), corresponding to the BN structure in Fig. 2.1, has been shown to apply to various sensor types [1, 34]. When sensor measurements are available from a target $\mathcal{T}_i$, a junction-tree BN inference algorithm [67] computes the posterior PMF $P(F_i, y_i \mid \mathcal{E}_i)$ from the BN CPTs in (2.9), and from the evidence $\mathcal{E}_i$, which includes the values of the measurements M_i, and the values of known operating (S_i) and environmental conditions (E_i). The unknown target features and classification are estimated as the values with the highest posterior probability, i.e., $\hat{T}_i, \hat{y}_i = \text{argmax}_{F_i^j, y_i^l} P(F_i = F_i^j, y_i = y_i^l \mid \mathcal{E}_i)$. And, the posterior probability $P(\hat{F}_i, \hat{y}_i \mid \mathcal{E}_i)$ is known as the confidence level (CL). The BN models of multiple and heterogeneous sensors can be used in combination with the Dempster-Shafer rule of evidence combination to perform feature-level sensor fusion [1]. Also, in this dissertation, the factorization (2.9) is used to compute the information value of a measurement set, $V(\tau)$ in (4.1), before the measurements become available.

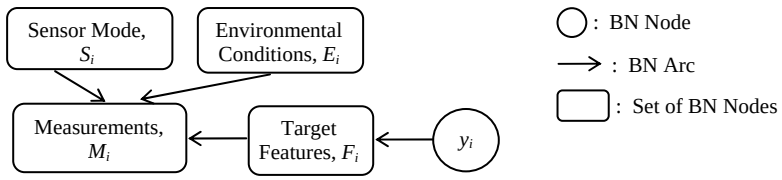


FIGURE 2.1: Typical structure of a BN sensor model [1].

2.3 Probabilistic Roadmap Methods

Probabilistic roadmap methods (PRMs) are a class of randomized motion planning algorithms that have recently received considerable attention because they are capable of handling problems with many degrees of freedom, or large workspaces with many obstacles, for which other motion planning methods are computationally infeasible [38–41]. So far, these methods have been applied to classical motion planning, in order to plan the path of a robot with geometry $\mathcal{A}$ that connects $\mathbf{q}_0$ to $\mathbf{q}_f$ by

a minimum distance, avoiding collisions with the set of obstacles B in $\mathcal{W}$. In this dissertation, PRM is modified to solve the sensor path planning problem, thereby also accounting for the sensor's field-of-view geometry $\mathcal{S}$, and the set of targets T in $\mathcal{W}$ to find the robotic sensor path τ.

PRM planners sample the free configuration space $\mathcal{C}_{free}$ and construct a roadmap graph by means of a *learning phase*. Subsequently, a collision-free path from $\mathbf{q}_0$ to $\mathbf{q}_f$ is determined from the roadmap by a so-called *query phase* [41]. The roadmap is an undirected graph $G = (L, A)$, comprised of a set of nodes or *milestones* $L = \{\mathbf{c}_1, \ldots, \mathbf{c}_{N_m}\}$, and a set of arcs $A = \{(\mathbf{c}_i, \mathbf{c}_j) |\ \mathbf{c}_i, \mathbf{c}_j \in L\}$. Every milestone $\mathbf{c}_i \in L$ represents a value of the robot configuration $\mathbf{q}$ sampled from $\mathcal{C}_{free}$ using a probability density function (PDF). Every arc $(\mathbf{c}_i, \mathbf{c}_j) \in A$ represents a simple local path, typically a straight line, that connects $\mathbf{c}_i$ and $\mathbf{c}_j$ in $\mathcal{C}_{free}$, and is determined by an extremely fast local planner. Assuming the roadmap properly represents the connectivity of $\mathcal{C}_{free}$, multiple queries can be used in the second stage to construct a path from $\mathbf{q}_0$ to $\mathbf{q}_f$ by concatenating several feasible paths in G, in order of increasing distance from nodes $\mathbf{q}_0$ and $\mathbf{q}_f$, until they are successfully connected. A smoothing algorithm can then be applied to obtain a more natural and shorter path by means of geometric operations [68].

The main difficulties in constructing appropriate roadmaps for PRMs are providing coverage of wide-open regions of $\mathcal{C}_{free}$, and representing the connectivity of $\mathcal{C}_{free}$ even in the presence of narrow passages. The use of uniform and Gaussian distributions have been proposed in [41] and in [40], respectively, in order to generate milestones that would properly cover wide-open regions of $\mathcal{C}_{free}$. More recently, an hybrid strategy combining a bridge test with a Gaussian probability density function along narrow passages, and a uniform probability density function in wide-open regions of $\mathcal{C}_{free}$ was proposed in [38], in order to construct appropriate roadmaps for workspaces with both characteristics. The bridge test is based on the observation

that a narrow passage in $\mathcal{C}$ has at least one restricted direction between C-obstacles that can be identified by a short straight-line segment or *bridge* with endpoints on two C-obstacles. Let $\mathbf{b}$ and $\mathbf{b}'$ be two random variables representing the two endpoints of a bridge in $\mathcal{C}$. Since bridges must connect C-obstacles, $\mathbf{b}$ is sampled from a uniform distribution $f(\mathbf{b})$ over $\mathcal{CB}$ that is zero in $\mathcal{C}_{free}$, and $\mathbf{b}'$ is sampled from the conditional probability density function,

$$f(\mathbf{b}' \mid \mathbf{b}) = \lambda_{\mathbf{b}}(\mathbf{b}')I(\mathbf{b}')/Z_{\mathbf{b}} \tag{2.10}$$

normalized by $Z_{\mathbf{b}} \equiv \int_{\mathcal{C}} \lambda_{\mathbf{b}}(\mathbf{b}')I(\mathbf{b}')d\mathbf{b}'$. $\lambda_{\mathbf{b}}$ is a multivariate Gaussian probability density function that is radially symmetric, and has a standard deviation that depends on the width of the bridge length (Section 4.4.2). $I(\mathbf{b})$ is a binary function that equals 1 when $\mathbf{b} \in \mathcal{CB}$, and 0 otherwise. Then, as shown in [38], milestones can be obtained with a higher frequency inside narrow passages by sampling the probability density function,

$$\pi_G(\mathbf{q}) = \int_{\mathcal{C}} f(\mathbf{b}' \mid \mathbf{b})f(\mathbf{b})d\mathbf{b} = \int_{\mathcal{CB}} \lambda_{\mathbf{b}}(2\mathbf{q} - \mathbf{b})I(2\mathbf{q} - \mathbf{b})/Z_{\mathbf{b}}d\mathbf{b} \tag{2.11}$$

obtained by choosing the desired configuration $\mathbf{q}$ as the midpoint on the bridge from $\mathbf{b}$ to $\mathbf{b}'$, such that $\mathbf{b}' = 2\mathbf{q} - \mathbf{b}$. In order to also include milestones from wide-open regions of $\mathcal{C}$, a uniform probability density function π_U is defined over $\mathcal{C}_{free}$, and the hybrid sampling strategy,

$$\pi_H = v\pi_G + (1 - v)\pi_U \tag{2.12}$$

is used to cover both wide-open regions and narrow passages in $\mathcal{C}_{free}$. The user-defined parameter $0 \leq v \leq 1$ is chosen to emphasize either distributions in the mixture (Section 4.4.2).

In this dissertation, a new probability density function is presented for sampling milestones based on the geometry and information value of the targets in $\mathcal{W}$. A

new hybrid sampling strategy (Section 4.2) is derived from prior measurements and environmental information, and from the probabilistic sensor model reviewed in the next section. Then, the profit of information, (4.1), is used in place of the classical distance metric [39] in the query phase (Section 4.3), to determine the path of maximum information profit in the roadmap.

2.4 Background on Potential Field

The potential field method is a robot motion planning technique that utilizes an artificial potential function to find the obstacle-free path of shortest distance in an Euclidian workspace. The obstacles and the goal configuration, are considered as sources to construct a potential function U which represents the characteristics of the configuration space. Although different approaches have been utilized to generate U [51–53,69], the potential function always consists of two components, the attractive potential U_{att} generated by the goal configuration, and the repulsive potential U_{rep}, generated by the obstacles. The total potential is given by,

$$U(\mathbf{q}) = U_{att}(\mathbf{q}) + U_{rep}(\mathbf{q}) \tag{2.13}$$

where $\mathbf{q}$ is any configuration in $\mathcal{C}$. The force applied on the robot is proportional to the negative gradient of U,

$$\nabla U(\mathbf{q}) = [\frac{\partial U(\mathbf{q})}{\partial q_1}, \frac{\partial U(\mathbf{q})}{\partial q_2}, \ldots, \frac{\partial U(\mathbf{q})}{\partial q_n}]^T \tag{2.14}$$

where $\mathbf{q} = [q_1 \ q_2 \ \ldots \ q_n]^T \in \mathbb{R}^n$.

As shown in [50], the repulsive potential can be represented as,

$$U_{rep}(\mathbf{q}) = \begin{cases} \frac{1}{2}\eta(\frac{1}{\rho(\mathbf{q})} - \frac{1}{\rho_0})^2 & \text{if} \quad \rho(\mathbf{q}) \leq \rho_0 \\ 0 & \text{if} \quad \rho(\mathbf{q}) > \rho_0 \end{cases} \tag{2.15}$$

where η is a scaling factor, $\rho(\mathbf{q})$ is the distance between the robot and the nearest obstacle in Euclidean space, and ρ_0 is a constant parameter that is chosen by the user. The attractive potential is given by,

$$U_{att}(\mathbf{q}) = \frac{1}{2}\varepsilon\rho^2_{goal}(\mathbf{q}) \tag{2.16}$$

where ε is a scaling factor, and $\rho_{goal}(\mathbf{q})$ is the distance between the robot and the goal configuration. In (2.15) and (2.16), only the obstacle closest to $\mathbf{q}$ is considered to generate $U_{rep}(\mathbf{q})$, and the target is assumed to be a single point in $\mathcal{C}_{free}$. This makes the potential function difficult to be updated online when new obstacles and targets are sensed during the path execution, because for each value of $\mathbf{q}$, the potential needs to update by computing its distance from the closest obstacle and target.

In this dissertation, a novel potential function is presented that takes into account the geometries of the sensor's FOV and of the targets, as well as the information value of the targets. The proposed potential function is used for online sensor network path planning in Chapter 5.

3

Comparison of Information Functions for Sensor Classification

In this chapter, various information functions are utilized to derive the information value of measuring a group of cells. Information-driven search strategy based on these information functions is demonstrated in a canonical sensor planning problem with five sensor models and their performance is compared to other existing search strategies.

3.1 Canonical Sensor Planning Problem

As discussed in Chapter 1, in many sensor applications, the state vector, the measurements, and the sensor characteristics are random variables. The sensor measurement model can be represented by the joint PMF $p(\mathbf{Z}^k, \mathbf{X}^k, \lambda^k)$, which typically is known in terms of the factorization,

$$p(\mathbf{Z}^k, \mathbf{X}^k, \lambda^k) = p(\mathbf{Z}^k \mid \mathbf{X}^k, \lambda^k)p(\mathbf{X}^k)p(\lambda^k) \tag{3.1}$$

and assumes $\mathbf{X}^k$ and λ^k are independent [1, 6, 25]. Both models (1.1) and (3.1) are time invariant, and are obtained in discrete time, indexed by $k = 1, \ldots, f$.

The probabilistic model in (3.1) assumes that $\mathbf{Z}^k$, $\mathbf{X}^k$, and λ^k are discrete random variables with finite ranges $\mathcal{X}$, $\mathcal{Z}$, and Λ, respectively. Continuous random variables can be considered using an analogous model based on the joint probability density function (PDF) [22]. The conditional PMF $p(\mathbf{Z}^k \mid \mathbf{X}^k, \lambda^k)$ is sometimes also referred to as sensor model, and is obtained from the physical principles underlying the measurement process. The priors $p(\mathbf{X}^k)$ and $p(\lambda^k)$ are computed from prior environmental information when available or, otherwise, are assumed to be uniformly distributed. Various sensors, including infrared, ground penetrating radars, and synthetic aperture radars, have been modeled using (3.1) for demining, surveillance, and radar target tracking applications [1, 21, 22, 27, 34]. In the remainder of this chapter, the joint PMF (3.1) is considered as the sensor model. The problem dimensions are chosen as $n = r = 1$, and $\wp = 3$.

Consider a sensor modeled by (1.1) or (3.1) that is implemented for the purpose of classifying multiple targets confined to a discrete set of cells $\mathcal{K} = \{\kappa_1, \ldots, \kappa_c\}$, with $c \geq f$. The sensor planning problem considered in this chapter consists of selecting one cell from $\mathcal{K}$ at every time $k = 1, \ldots, f$, such that the rate of correct target classification is maximized and the rate of false alarms is minimized. It is assumed that there is at most one target in each cell, and that targets do not move and do not change over time. Then, the state of the i^{th} cell κ_i, denoted by X_i, can be used to represent both the presence and classification of a target in κ_i. The range $\mathcal{X}$ contains a value representing an empty cell, and a set of values representing all possible target types, including a high-risk target type denoted by x^r. The measurement variable associated with κ_i, denoted by Z_i, is a discrete random variable with finite range $\mathcal{Z}$, that is unknown *a priori*. If the sensor obtains measurements from κ_i at time k, then $Z^k = Z_i = z^k$, $X^k = X_i$, and the measurement value z^k is known thereafter. Since measurements are imperfect and random, the actual value of X_i remains unknown, but can be estimated from z^k, as explained in Section 3.3. Over

time the sensor can obtain multiple measurements from the same cell in order to improve its estimate. Therefore, at every time k, the sensor planning algorithm must decide which cell to measure before z^k is known, and based on the set of measurements $\mathcal{M}^{k-1} = \{z^1, \ldots, z^{k-1}\}$ obtained up to time $(k-1)$.

This problem formulation is inspired by canonical sensor planning problems previously presented in [21, 32], and can be used to manage sensors in demining, and sensing-and-pursuit applications, as shown in [6, 7, 10, 70]. For example, a set $\mathcal{K}$ that obeys the aforementioned assumptions can be obtained from multiple geometric targets located in an obstacle-populated workspace, using the approximate cell decomposition method developed in [6]. This formulation can also be used in the case of moving targets by viewing the cells as state variables at different time instants, as shown in [7, 10, 22, 70]. Then, the value of information associated with a cell $\kappa_i \in \mathcal{K}$ at time k can be estimated using the information functions presented in the next section.

3.2 Information Value Functions for Sensor Planning

Information theoretic functions are a natural choice for representing the value of information because they measure the absolute or relative information content of probability mass (or density) functions. However, as shown in Chapter 2, computing these functions requires knowledge of the PMFs representing the prior and posterior belief state. Since in sensor planning problem the posterior belief state typically is unknown, in this section, we present a general approach for utilizing information theoretic functions to estimate the information value of a cell or target prior to obtaining the corresponding sensor measurements and posterior belief. Although it is assumed that the sensor parameters are known *a priori*, the approach can be easily extended to the case in which they also must be selected by the sensor manager by adjoining them to Z_i.

Suppose the sensor manager is considering obtaining an unknown measurement Z_i from κ_i at time k, given all past measurements $\mathcal{M}^{k-1}$. Then, the information value of κ_i can be represented by the change in belief state brought about by Z_i, as measured by the α-divergence in (2.3). At time k, the change between the prior belief state, $p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)$, and the posterior belief state, $p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)$, can be estimated by taking the expectation with respect to Z_i, denoted by $\mathbb{E}_{Z_i}$. Then, from (2.3), the information value can be represented by the expected α-divergence, defined as,

$$\begin{aligned}
&\hat{\varphi}_{D_\alpha}(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \\
&\equiv \mathbb{E}_{Z_i}\left\{D_\alpha[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i) \parallel p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)]\right\} \\
&= \sum_{z_j \in \mathcal{Z}} D_\alpha[p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i) \parallel p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \qquad (3.2)
\end{aligned}$$

By taking the expectation with respect to Z_i, the measurement value z^k is no longer needed, and $\hat{\varphi}_{D_\alpha}$ can be computed from $\mathcal{M}^{k-1}$ and the sensor model, as explained below.

As shown in [22], measurements that are obtained at different time instants can be assumed to be conditionally independent given the state, i.e.,

$$p(z^{k-1} \mid X_i, z^{k-2}, \ldots, z^1, \lambda_i) = p(z^{k-1} \mid X_i, \lambda_i) \qquad (3.3)$$

As a result, an efficient rule can be derived for updating the belief state iteratively over time using the sensor model (3.1) in combination with Bayes' rule. Suppose a measurement z^{k-1} is obtained from cell $\kappa_i \in \mathcal{K}$ at time $(k-1)$. Then, the PMF of

X_i given $\mathcal{M}^{k-1}$ and λ_i can be updated by the rule,

$$\begin{aligned} p(X_i \mid \mathcal{M}^{k-1}, \lambda_i) &= p(X_i \mid z^{k-1}, \mathcal{M}^{k-2}, \lambda_i) \\ &= \frac{p(z^{k-1} \mid X_i, \mathcal{M}^{k-2}, \lambda_i) p(X_i \mid \mathcal{M}^{k-2}, \lambda_i)}{p(z^{k-1} \mid \mathcal{M}^{k-2}, \lambda_i)} \\ &= \frac{p(z^{k-1} \mid X_i, \lambda_i) p(X_i \mid \mathcal{M}^{k-2}, \lambda_i)}{\sum_{x_\ell \in \mathcal{X}} p(z^{k-1} \mid X_i = x_\ell, \lambda_i) p(X_i = x_\ell \mid \mathcal{M}^{k-2}, \lambda_i)} \end{aligned} \tag{3.4}$$

where, $p(X_i \mid \mathcal{M}^{k-2}, \lambda_i)$ is known from the previous time step $(k-2)$, and $p(z^{k-1} \mid X_i, \lambda_i)$ is known from (3.1). As additional measurements are obtained at subsequent time steps, (3.4) can be implemented iteratively by updating the time and cell index accordingly. Finally, the posterior belief inside the expectation in (3.2) is computed by applying Bayes' rule for every value $z_j \in \mathcal{Z}$:

$$\begin{aligned} &p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i) \\ &= \frac{p(z_j \mid X_i, \lambda_i) p(X_i \mid \mathcal{M}^{k-2}, \lambda_i)}{\sum_{x_\ell \in \mathcal{X}} p(z_j \mid X_i = x_\ell, \lambda_i) p(X_i = x_\ell \mid \mathcal{M}^{k-2}, \lambda_i)} \end{aligned} \tag{3.5}$$

The expected discrimination gain (EDG), originally proposed in [21], can be derived by taking the expectation of the Kullback-Leibler divergence defined in (2.4), i.e.:

$$\begin{aligned} &\hat{\varphi}_D(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \\ &\equiv \mathbb{E}_{Z_i} \left\{ D[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i) \parallel q(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \right\} \\ &= \sum_{z_j \in \mathcal{Z}} D[p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i) \parallel q(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \\ &\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \end{aligned} \tag{3.6}$$

It can be seen that, by this approach, EDG can be computed from the same PMFs in (3.4)-(3.5) used to compute $\hat{\varphi}_{D_\alpha}$.

As shown in [6], an information value function based on conditional mutual information can be used to represent the reduction in uncertainty in X_i, due to the

knowledge of Z_i, when $\mathcal{M}^{k-1}$ and λ_i are given. Based on the definition in (2.5), computing the conditional mutual information $I(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i)$ requires computing the entropy $H(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)$, which in turn requires knowledge of Z_i (as pointed out in [18]). Therefore, a more suitable measure for sensor planning is the expected conditional mutual information (EER), defined as

$$\begin{aligned}
&\hat{\varphi}_I(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \equiv \mathbb{E}_{Z_i}\left\{I(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i)\right\} \\
&= H(X_i \mid \mathcal{M}^{k-1}, \lambda_i) - \mathbb{E}_{Z_i}\left\{H(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)\right\} \\
&= H(X_i \mid \mathcal{M}^{k-1}, \lambda_i) - \sum_{z_j \in \mathcal{Z}} H(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i) \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i)
\end{aligned} \tag{3.7}$$

Where, the entropy $H(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i)$ is computed from (3.5), using (2.1). The expected Cauchy-Schwartz information function, derived from the CS measure in (2.6), and defined as,

$$\begin{aligned}
&\hat{\varphi}_C(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \\
&\equiv \mathbb{E}_{Z_i}\left\{C[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i), p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)]\right\} \\
&= \sum_{z_j \in \mathcal{Z}} C[p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i), p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i)
\end{aligned} \tag{3.8}$$

can be used to obtain an alternative measure of the distance between the prior and the posterior belief state for a cell κ_i, prior to obtaining the measurement Z_i.

Although they have not been previously applied to sensor planning, the quadratic entropy and information potential are considered here because quadratic forms are well-suited to numerical optimization. By viewing each of these two functions as a measure of the uncertainty associated with a distribution, the information value of κ_i can be represented by the expected information potential gain (EIPG), defined

as,

$$\begin{aligned}
&\hat{\varphi}_V(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \\
&\equiv \mathbb{E}_{Z_i}\left\{V[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)] - V[p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)]\right\} \\
&= \mathbb{E}_{Z_i}\left\{V[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)]\right\} - V[p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \\
&= \sum_{z_j \in \mathcal{Z}} V[p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i)] p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \\
&- V[p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] \qquad (3.9)
\end{aligned}$$

and derived from the information potential in (2.7). The expected quadratic entropy reduction (EQER) is similarly defined as,

$$\begin{aligned}
&\hat{\varphi}_{R_2}(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i) \\
&\equiv \mathbb{E}_{Z_i}\left\{R_2[p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] - R_2[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)]\right\} \\
&= R_2[p(X_i \mid \mathcal{M}^{k-1}, \lambda_i)] - \mathbb{E}_{Z_i}\left\{R_2[p(X_i \mid Z_i, \mathcal{M}^{k-1}, \lambda_i)]\right\} \\
&= R_2[p(X_i \mid \mathcal{M}^{k-1})] - \sum_{z_j \in \mathcal{Z}} R_2[p(X_i \mid Z_i = z_j, \mathcal{M}^{k-1}, \lambda_i)] \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \qquad (3.10)
\end{aligned}$$

and, like EIPG, can be computed from the PMFs in (3.4)-(3.5). It can be seen from (3.9) and (3.10) that optimizing the EIPG is not equivalent to optimizing EQER.

In the next sections, the above information value functions are utilized to select the optimal sequence of cells for the canonical sensor problem in Section 3.1, and compared to other search strategies previously presented in the literature.

3.3 Search Strategies

A greedy information-driven search (IS) strategy for solving the sensor planning problem presented in Section 3.1 consists of selecting the sequence of cells with the maximum expected information value. Assuming the expected information value $\hat{\varphi}$

can be represented by one of the functions derived in Section 3.2, at every time k, the cell κ_l with index,

$$l = \arg\max_i \{\hat{\varphi}(X_i; Z_i \mid \mathcal{M}^{k-1}, \lambda_i)\} \tag{3.11}$$

is selected from $\mathcal{K}$. After the measurement value $Z_l = z^k$ is obtained, the measurement set is updated by letting $\mathcal{M}^k = \{\mathcal{M}^{k-1}, z^k\}$, and the posterior PMF of X_l is updated using (3.4). The IS strategy is applied iteratively over time, until $k = f$. For comparison, direct-search, alert-confirm search, task-driven, and log-likelihood-ratio search strategies, which have been shown to outperform other approaches, such as, index rule, and sequential-probability-ratio test in [32] are also implemented on the canonical sensor problem in Section 3.1.

Direct search (DS) consists of selecting cells in the order in which they are provided in $\mathcal{K}$, obtaining only one measurement from each cell [32]. Alert-confirm (AS) is a sensor management approach implemented in multitarget radar systems which consists of selecting cells in the order in which they are provided in $\mathcal{K}$ until an "alert" is obtained. The alert triggers a "confirm" cycle that obtains additional measurements until a desired level of confidence (CL), such as a desired signal-to-noise ration (SNR), is achieved, or until the probability of detection exceeds that of false alarm [32, 33]. Only when the confirm cycle is completed, the sensor moves on to the next cell. In this chapter, the CL for the AS strategy is represented by a desired belief state or posterior PMF, defined as,

$$\mathrm{CL}_i \equiv \max_{x_j} p(X_i = x_j \mid \mathcal{M}^{k-1}, \lambda_i) \tag{3.12}$$

and is required to be equal to 0.9. Therefore, at time k the sensor obtains a measurement from κ_i if $\mathrm{CL}_i < 0.9$. Otherwise, the sensor moves to κ_{i+1}, and obtains measurements from κ_{i+1} until $\mathrm{CL}_{i+1} \geq 0.9$. Although AS can lead to good sensor

performance, its key disadvantage is that it cannot be used to minimize time or energy consumption.

Task-driven (TS), or Bayes-risk, search strategies have been recently proposed for applications in which a sensor is deployed to perform a single task, such as to find mines or minimize tracking errors, and base the management decisions on a heuristic criterion, such as the associated risk level [27, 42]. These applications can be considered by introducing a high-risk state value x^r in the canonical sensor problem, which represents a high-risk target class (such as a mine) or target location. Then, the TS strategy selects the cells with the maximum *a-posteriori* probability for the value x^r, optimizing the risk associated with declaring $X_i = x^r$ [27,42]. Since the *a-posteriori* probability of x^r is unknown at the time of the sensor decision, it is estimated as follows,

$$\hat{P}_r^k(X_i) \equiv \mathbb{E}_{Z_i}\left[p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i, Z_i)\right] \tag{3.13}$$

similarly to the approach proposed in [27]. Then, at time k, the TS strategy selects the cell κ_l, with index,

$$l = \arg\max_j \{\hat{P}_r^k(X_i)\} \tag{3.14}$$

from $\mathcal{K}$. Since $\hat{P}_r^k(X_i)$ is independent of Z_i, (3.14) can be simplified as follows,

$$
\begin{aligned}
l &= \arg\max_i \left\{ \mathbb{E}_{Z_i} \left[p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i, Z_i) \right] \right\} \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i, Z_i = z_j) \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} \frac{p(X_i = x^r, \mathcal{M}^{k-1}, \lambda_i, Z_i = z_j)}{p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) p(\mathcal{M}^{k-1}, \lambda_i)} \\
&\times p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} p(X_i = x^r, Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i) \\
&= \arg\max_i p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i)
\end{aligned} \tag{3.15}
$$

and the TS strategy selects cells in the order of decreasing probability for the high-risk state x^r.

The log-likelihood-ratio (LLR) criterion has been successfully implemented for the evaluation of track formation hypotheses in multiple target tracking applications [31, 33]. When two possible hypothesis can be used to explain the data, the LLR is proportional to the ratio between the likelihood of the data given the first hypothesis, over the likelihood of the data given the second hypothesis. Typically, the likelihood is multiplied by the prior probability of the hypothesis, such that the LLR can be computed recursively using Bayes' rule [33]. In this chapter, the LLR definition in [33] is extended to non-binary state variables with a high-risk value x^r, as follows:

$$
\begin{aligned}
L^k(X_i) \;&\equiv\; \ln \left[\frac{p(Z_i \mid X_i = x^r) p(X_i = x^r)}{p(Z_i \mid X_i \neq x^r) p(X_i \neq x^r)} \right] \\
&= \ln \left[\frac{p(Z_i \mid X_i = x^r) p(X_i = x^r)}{\sum_{x_j \in \mathcal{X}, x_j \neq x^r} p(Z_i \mid X_i = x_j) p(X_i = x_j)} \right]
\end{aligned} \tag{3.16}
$$

Then, since a cell κ_i must be selected before the measurement Z_i and its likelihoods

are known, a LLR search (LLRS) strategy is developed by taking the expectation of (3.16), and selecting the cell κ_l with the highest expected LLR, as shown in (3.17). It can be seen from (3.17) that LLRS selects cells in the order of decreasing log ratio of the expected posterior probability of x^r, divided by the expected posterior probability of all other state values in $\mathcal{X}$.

$$
\begin{aligned}
l &= \arg\max_i \mathbb{E}_{Z_i}\left\{\ln\left[\frac{p(Z_i \mid X_i = x^r, \mathcal{M}^{k-1}, \lambda_i)p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i)}{p(Z_i \mid X_i \neq x^r, \mathcal{M}^{k-1}, \lambda_i)p(X_i \neq x^r \mid \mathcal{M}^{k-1}, \lambda_i)}\right]\right\} \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i)\times \\
&\ln\left[\frac{p(Z_i = z_j \mid X_i = x^r, \mathcal{M}^{k-1}, \lambda_i)p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i)}{p(Z_i = z_j \mid X_i \neq x^r, \mathcal{M}^{k-1}, \lambda_i)p(X_i \neq x^r \mid \mathcal{M}^{k-1}, \lambda_i)}\right] \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i)\times \\
&\ln\left[\frac{p(Z_i = z_j, X_i = x^r, \mathcal{M}^{k-1}, \lambda_i)p(\mathcal{M}^{k-1}, \lambda_i)}{p(Z_i = z_j, X_i \neq x^r, \mathcal{M}^{k-1}, \lambda_i)p(\mathcal{M}^{k-1}, \lambda_i)}\right] \\
&= \arg\max_i \sum_{z_j \in \mathcal{Z}} p(Z_i = z_j \mid \mathcal{M}^{k-1}, \lambda_i)\times \\
&\ln\left[\frac{p(X_i = x^r \mid \mathcal{M}^{k-1}, \lambda_i, Z_i = z_j)}{\sum_{x_j \in \mathcal{X}, x_j \neq x^r} p(X_i = x_j \mid \mathcal{M}^{k-1}, \lambda_i, Z_i = z_j)}\right]
\end{aligned}
\tag{3.17}
$$

3.4 Sensor Decision Rules and Performance

The sensor decision rule refers to the criterion used to decide or estimate the state value of a cell after the sensor measurement is obtained. Although this criterion does not affect the search strategy because it is implemented *a posteriori*, it affects the sensor performance. As reviewed in [31], a number of decision rules may be employed to decide what value of X_i to accept, based on the posterior belief state or PMF $p(X_i \mid \mathcal{M}^k, \lambda_i)$, computed from (3.4) using the latest measurement z^k. The Neyman-Pearson rule accepts a state value only if its likelihood ratio is greater than a

desired significance level. The minimax and Bayes cost functions utilize user-defined weights to quantify the risk or cost of choosing one state value over the others based on its likelihood and posterior probability, respectively. When these weights are set equal to one, the minimax and Bayes cost functions reduce respectively to the maximum likelihood and maximum *a-posteriori* rules implemented in this chapter.

The maximum *a-posteriori* rule accepts the state value $x^* \in \mathcal{X}$, if its probability is greater than the posterior probability of any other value given the data, i.e.:

$$\begin{aligned} \hat{X}_i = x^*, \quad \text{iff} \quad p(X_i = x^* \mid \mathcal{M}^k, \lambda_i) \geq p(X_i = x_\ell \mid \mathcal{M}^k, \lambda_i), \\ \forall x_\ell \in \mathcal{X}, \quad x_\ell \neq x^* \end{aligned} \tag{3.18}$$

When $\mathcal{M}^k$ contains no measurements about a cell κ_i, the prior is used in place of the posterior probability in (3.18). The maximum-likelihood (MLE) rule accepts the state value $x^* \in \mathcal{X}$ if its likelihood is greater than that of any other value given the data, i.e.:

$$\begin{aligned} \hat{X}_i = x^*, \quad \text{iff} \quad p(z^k \mid X_i = x^*) \geq p(z^k \mid X_i = x_\ell), \\ \forall x_\ell \in \mathcal{X}, \quad x_\ell \neq x^* \end{aligned} \tag{3.19}$$

It can be seen from (3.19) that the MLE rule can only be applied to cells for which at least one measurement has been obtained.

For the canonical sensor problem considered in this chapter, the sensor performance is characterized by the rate of correct classification and the rate of false alarms. Let n denote the total number of cells with a state value that is estimated correctly, and n_r denote the number of cells with the high-risk state value x^r that is estimated correctly. The number of false alarms, n_{fa}, represents the number of cells that are empty but are incorrectly declared to contain a target by the decision rule. Then, at any time k, the rate of correct classification is,

$$F_c(k) = \frac{n(k)}{c} \tag{3.20}$$

where c is the total number of cells in $\mathcal{K}$. The rate of correct classification for high-risk targets at time k is,

$$F_r(k) = \frac{n_r(k)}{c_r} \tag{3.21}$$

where c_r is the number of cells in $\mathcal{K}$ with state value x^r. The frequency of false alarms at time k is,

$$F_{fa}(k) = \frac{n_{fa}(k)}{c_e} \tag{3.22}$$

where c_e is the number of empty cells in $\mathcal{K}$.

3.5 Simulated Sensor Models

The comparative performance of the information functions, search strategies, and decision rules presented in the previous sections is investigated by simulating three types of prior distributions, and five sensor models. Four sensor models are simulated by generating the joint PMF in (3.1) from Bernoulli, binomial, Poisson, and mixture-of-binomials distributions, and one sensor model is simulated using the classical form in (1.1) using a nonlinear power law and Gaussian noise. The canonical sensor planning problem in Section 3.1 is simulated by generating the set $\mathcal{K}$ of c cells, with c_t targets, and $c = c_t + c_e$. The state of each cell has the range $\mathcal{X} = \{x_1, x_2, x_3, x_4\}$, where x_1 and x_2 denote two types of empty cells (e.g., clear terrain and clutter), x_3 and x_4 denote two types of targets, and the high-risk value is $x^r = x_4$. The sensor characteristics and environmental conditions in κ_i are represented by the random vector $\lambda_i = [\alpha_i \ \beta_i \ \gamma_i]^T$. Each random element in λ_i is assumed to have three possible values, thus the range Λ of λ_i has nine possible values.

The sensor model in (3.1) represents the joint probability of the sensor measurements in $\mathcal{Z}$, the target state values in $\mathcal{X}$, and the sensor and environmental characteristics in Λ. In practice, this joint probability distribution is determined

from the physical processes underlying the sensor measurements, and by the nature of the targets and environmental conditions in the region of interest [1]. In this chapter, (3.1) is simulated by means of well-known probability distributions, in order to conduct a comparative study that is independent of the sensing application. Let $\theta \in [0,\ 1]$ denote a parameter of the distribution that can be viewed as the influence of the cell state and environmental conditions on the probability of success of the sensor measurements. Then, θ can be modeled as a function of X_i and λ_i, i.e.,

$$\theta = g(X_i, \lambda_i) \equiv \eta_j\ \alpha_i^{a_{j1}} \beta_i^{a_{j2}} \gamma_i^{a_{j3}},\ X_i = x_j,\ \forall x_j \in \mathcal{X} \tag{3.23}$$

where, $\eta_j > 0$ and $A = \{a_{ij}\} \in \mathbb{R}^{4\times 4}$ are parameters that represent the effluence of X_i and λ_i on θ, respectively, and are shown in Appendix A.

Poisson models and Bernoulli trials have been used to model moving target detections by distributed sensor networks in [71–73]. A Bernoulli experiment has a random outcome that can take two mutually exclusive values, e.g., success or failure, and, when repeated N independent times, leads to a sequence of N Bernoulli trials [74]. Let the measurement Z_i denote the random variable associated with one Bernoulli trial. In the Bernoulli sensor model Z_i must be binary, thus all targets are assigned the high-risk value x^r. Then, the probability of a measurement's success ($z_j = 1$), e.g., target detection, and the probability of failure ($z_j = 0$), are used to generate the posterior PMF in the sensor model, based on the Bernoulli distribution

$$p(Z_i = z_j \mid \theta) = \theta^{z_j}(1-\theta)^{1-z_j},\ z_j = 0, 1. \tag{3.24}$$

In the binomial sensor model, the measurement Z_i denotes the number of observed successes in N Bernoulli trials, such that $\mathcal{Z} = \{z_1, \ldots, z_N\}$, where $N = 3$. Then, the posterior PMF in the binomial sensor model can be generated based on the binomial distribution,

$$p(Z_i = z_j \mid \theta) = \binom{N}{z_j} \theta^{z_j}(1-\theta)^{N-z_j},\ z_j = 0, \ldots, N, \tag{3.25}$$

where, X_i can take all four possible values in $\mathcal{X}$.

A Poisson process is a random experiment that generates a number of changes in a fixed interval, such as space or time, and whose probability can be described by an infinite series that converges to an exponential function [74]. Let the measurement Z_i denote the number of changes in the each interval, and let θ in (3.23) represent the parameter of the distribution. Then, assuming the number of changes in nonoverlapping intervals are independent, and $\theta > 0$, the Poisson sensor model can be generated from the Poisson distribution,

$$p(Z_i = z_j \mid \theta) = e^{-\theta} \frac{\theta^{z_j}}{z_j!}, \ z_j = 0, 1, 2, \ldots, \tag{3.26}$$

where, it can be shown that θ is equal to the expected number of changes in the process, or $\theta = \mathbb{E}(Z_i)$. Finally, the Bernoulli, binomial, and Poisson models are generated by substituting (3.23) in (3.24), (3.25), and (3.26), respectively, for all $z_j \in \mathcal{Z}$.

A more complex sensor model is obtained by means of a mixture of distributions, which compounds multiple PMFs using positive mixing proportions or weights [74]. Mixture models are used in a variety of applications, ranging from classification to statistical inference, and are reviewed comprehensively in [75]. Since the random variables considered in this chapter are discrete, we consider a mixture of two binomial distributions,

$$f(Y) = \sum_{l=1}^{2} w_l \binom{N_l}{Y} \theta_l^Y (1-\theta_l)^{N_l - Y}, \ 0 \leq w_l \leq 1, \sum_{l=1}^{2} w_l = 1, \tag{3.27}$$

formulated in terms of a discrete random variable Y with range $\mathcal{Y} = \cup_{l=1}^{2} \{0, \ldots, N_l\}$. The mixing proportions are w_1 and w_2, and the binomial parameters are θ_1 and θ_2. Let $\theta_1 = \theta = g(X_i, \lambda)$ in (3.23), with parameters shown in Appendix A. Also, let $\theta_2 = \sqrt{\theta_1}$, and $N_1 = 3$ and $N_2 = 5$. Then, the posterior PMF in the mixture-of-

binomials sensor model can be generated according to the distribution,

$$p(Z_i = z_j \mid \theta) = w\binom{N_1}{z_j}\theta^{z_j}(1-\theta)^{N_1-z_j} + (1-w)\binom{N_2}{z_j} \times \theta^{z_j/2}(1-\sqrt{\theta})^{N_2-z_j},\ 0 \leq w \leq 1,\ z_j = 0, \ldots, 5 \tag{3.28}$$

where, w is a user-defined weight that is set equal to 0.5.

A sensor model that is widely applicable and obeys the classical form (3.30) is an exponential power law that models the received isotropic energy generated by a constant target source level, and attenuated by the environment. This power law is commonly applied to acoustic, magnetic, and optical sensor measurements that are governed by linear wave propagation models [17, 76]. When the received signal exceeds the sensor's threshold, the target is detected and its distance from the sensor is computed using the sensor model and the known environmental conditions. We consider a multitarget scenario in which each target is located in one cell, say $\kappa_i \in \mathcal{K}$, and the state X_i represents the distance between the sensor and the target in κ_i. Then, the distance X_i can be estimated from a measurement Z_i obtained according to the power law,

$$Z_i = a_i||X_i||^{-\alpha_i} + \nu_i,\ i = 1, \ldots, c_t \tag{3.29}$$

and subject to additive, zero-mean Gaussian noise ν_i. Where, $\|\cdot\|$ denotes the L_2-norm, a_i is a known constant that depends on the target characteristics, and α_i is an attenuation coefficient that depends on the environmental conditions. In this chapter, it is assumed that X_i is a scalar, $a_i = 10$, and $\alpha_i = 0.3$ for all $\kappa_i \in \mathcal{K}$, obtaining the power law in Fig. 3.1 when $\nu_i = 0$.

In order to compare the classical model results to those obtained for the probabilistic sensor models, the range of X_i is discretized into four possible values $\mathcal{X} = \{1, 2, 3, 4\}$. The zero-mean Gaussian noise ν_i is fully specified by its standard de-

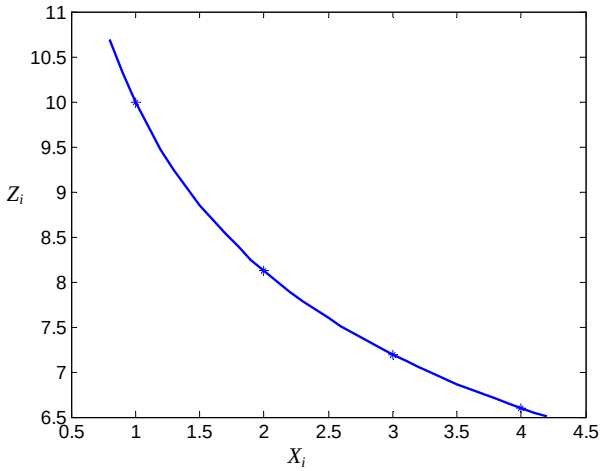


FIGURE 3.1: Received isotropic energy model as a function of distance from the target, for classical sensor model with $\nu_i = 0$.

viation σ_i. Thus, the sensor and environmental parameters can be represented by $\lambda_i = [a_i \ \alpha_i \ \sigma_i]^T$. It follows that the posterior probability of the classical sensor measurements in κ_i, given X_i and λ_i, is

$$p(Z_i = z_j \mid X_i, \lambda_i) \ = \frac{1}{\sqrt{2\pi}\ \sigma_i}\ e^{-(z_j - a_i X_i^{-\alpha_i})^2/(2\sigma_i^2)}, \qquad z_j \in [0,\ 13] \tag{3.30}$$

Where, Z_i and λ_i are continuous random variables, and X_i is a discrete random variable. Thus, (3.30) is referred to as a normal mixed model [77]. In this chapter, three noise models are considered with standard deviations $\sigma_i = 1$, 3, and 5, for all $\kappa_i \in \mathcal{K}$. As an example, the posterior probability (3.30) is plotted in Fig. 3.2 for $\sigma_i = 1$.

The five sensor models are simulated as follows. When a cell κ_i is selected from $\mathcal{K}$ at time k, one measurement value z^k is sampled from the corresponding joint PMF in (3.1), given the values of X_i and λ_i in κ_i. The prior $p(X_i)$ in (3.1) represents the prior information available about the cells' state variables. In this chapter, three models of $p(X_i)$ are considered that are referred to as uninformative prior, informative prior, and informative prior with large noise. The uninformative prior is modeled as a

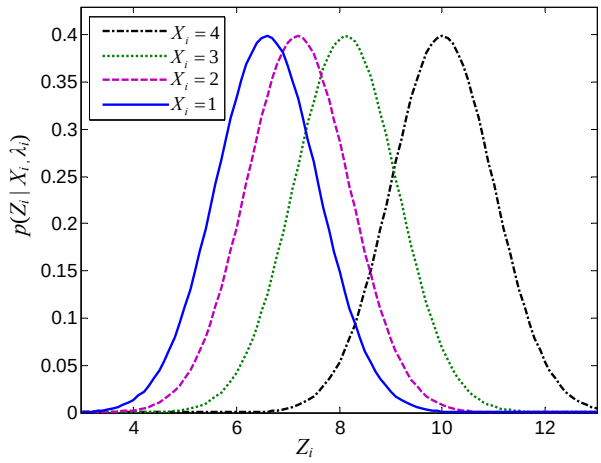


FIGURE 3.2: Posterior probability of classical sensor measurements for $\sigma_i = 1$.

uniform probability distribution over the range $\mathcal{X}$, indicating that any value of X_i is equally probable for any cell $\kappa_i \in \mathcal{K}$. The informative prior is generated by using the sensor to obtain one measurement per cell, i.e.:

$$p(X_i = x_\ell) = p(X_i = x_\ell \mid Z_1 = z^1, \ldots, Z_c = z^f), \quad \text{for } f = c, \text{ and } \forall x_\ell \in \mathcal{X}. \tag{3.31}$$

The informative prior with large noise represents imperfect expert knowledge or measurements obtained from a less accurate sensor. For the probabilistic sensor models in (3.24)-(3.28), this prior is generated by adding a random, uniformly-distributed error ϵ^k to every sensor measurement in (3.31), such that $Z_i = z^k + \epsilon^k$ for all $i, k = 1, \ldots, c$. For the classical sensor model (3.30), ϵ^k is sampled from a Gaussian distribution with zero-mean and standard deviation that is equal to $2\sigma_i$ for all $i, k = 1, \ldots, c$. The comparative results obtained by simulating the three priors and five sensor models described in this section are presented in the next section.

3.6 Numerical Simulations and Results

This section presents the results obtained by implementing the information theoretic functions presented in Section 3.2, and the search strategies and decision rules

presented in Section 3.3, on the canonical sensor problem presented in Section 3.1. The canonical sensor problem is simulated by generating a set $\mathcal{K}$ with $c = 1000$ cells as follows. In the Bernoulli sensor simulation, all $c_t = 500$ cells with targets are assigned the high-risk state value $x_4 = x^r$ (as explained in Section 3.5), 300 cells are assigned the value x_1 (empty), and 200 cells are assigned the value x_2 (clutter). In the binomial, Poisson, and binomial-mixture sensor simulations, 300 cells are assigned the value x_1 (empty), 200 cells are assigned the value x_2 (clutter), 200 cells are assigned the value x_3 (target), and 300 cells are assigned the value $x_4 = x^r$. In the classical sensor simulation, the state values represent the sensor's distance from the target (Section 3.5). Therefore, a target is assigned to every cell in $\mathcal{K}$, such that $c_t = c = 1000$, where 200 cells are assigned the distance x_1, 300 cells are assigned the distance x_2, 200 cells are assigned the distance x_3, and 300 cells are assigned the distance x_4. In all sensor simulations, the state values are assigned to the chosen number of cells in random order, by sampling the cell index using a uniform pseudorandom number generator [78].

In every simulation, the value of the parameter vector λ_i is assigned randomly to each cell, by sampling the uniform prior distribution $p(\lambda^k)$ over the range Λ. The probability $p(X_i)$ is not uniform and is unknown *a priori*. The three models of uninformative, informative, and large-noise informative priors (Section 3.5) are implemented separately to simulate cases in which prior information about X_i is either unavailable, available, or is available but very noisy, respectively. After the set of cells $\mathcal{K}$ is generated, the sensor measurements are obtained using the search strategies presented in Section 3.3. In order to investigate how the search performance varies as a function of time, k is varied from 0 to up to 3000 time steps. As soon as a measurement is obtained from a cell, it is processed using the belief-state update rule in (3.4), and the decision rules in Section 3.4.

The numerical results are organized as follows. For each sensor model, the rates

of correct classification (F_c), correct classification for high-risk targets (F_r), and false alarms (F_{fa}) are first evaluated using the IS strategy with each of the information functions presented in Section 3.2. The information function with the best performance is then used to compare the IS strategy to DS, AS, TS, and LLR strategies. For every sensor model and search strategy, a comparative study is conducted using the maximum *a-posteriori* and MLE decision rules, and the three models of priors. The results obtained from extensive numerical simulations are summarized and organized by sensor model in the next five subsections.

3.6.1 Bernoulli Sensor Model

The correct classification rate for high-risk targets, F_r, obtained by the IS strategy using different information functions, the informative prior, and the maximum *a-posteriori* and MLE rules is plotted in Figs. 3.3 and 3.4. It can be seen that the maximum *a-posteriori* rule significantly outperforms the MLE rule for all information functions. This comparison is representative of the numerical results obtained for other performance metrics (F_c and F_{fa}), and for other priors and sensor models, which are omitted here for brevity. Also, while it may be well-suited for sensor applications that require a very low rate of false alarms, the Neyman-Pearson rule was found to perform poorly compared to both the *a-posteriori* and MLE rules, because of its high rate of unclassified cells that do not meet the required significance level. Therefore, only the results obtained by the maximum *a-posteriori* decision rule are presented hereon in the chapter.

It can also be seen from Fig. 3.3 that the quadratic-entropy information function, $\hat{\varphi}_{R_2}$, leads to the best rate of correct classification F_r in the presence of an informative prior. On the other hand, as shown in Fig. 3.5, $\hat{\varphi}_{R_2}$ leads to the highest rate of false alarms compared to other information functions. In the Bernoulli sensor simulation, all information functions were found to display approximately the same value of F_c

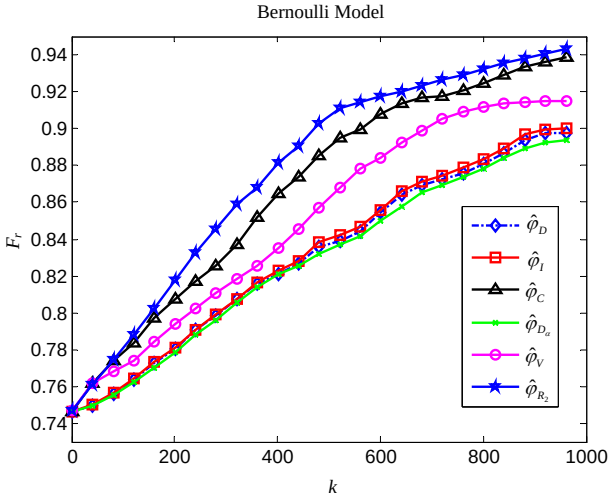


FIGURE 3.3: Correct classification rate for high-risk targets obtained with the maximum *a-posterior* decision rule, and an informative prior.

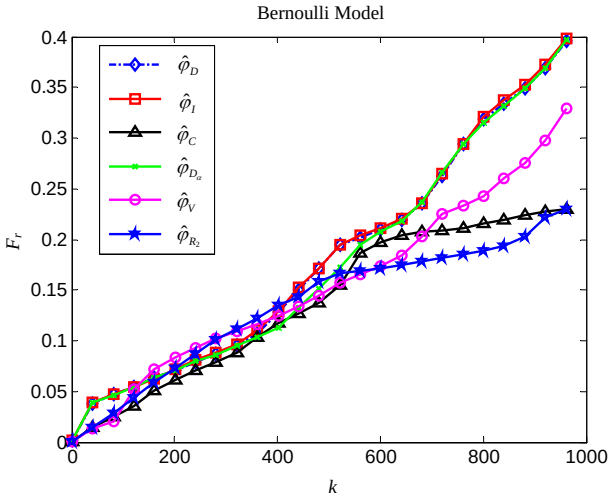


FIGURE 3.4: Correct classification rate for high-risk targets obtained with the MLE decision rule, and an informative prior.

for all $0 \leq k \leq f$. Therefore, information functions that achieved high values of F_r, performed poorly in terms of F_{fa}, and viceversa. The information function $\hat{\varphi}_I$, based on mutual information, provides a good compromise between F_r and F_{fa}. Therefore, for the Bernoulli sensor model, the IS strategy is compared to direct search, alert-confirm search, task-driven search, and log-likelihood-ratio search, using $\hat{\varphi}_I$.

As shown in Fig. 3.6, in the presence of an informative prior, the IS strategy was found to achieve the highest rate of correct classification, and the highest rate of correct classification for high-risk targets (not shown for brevity). The lowest false-alarm rate was obtained by the AS, TS, and LLRS strategies, because they obtain multiple measurements from few selected cells with low uncertainty, until they achieve high confidence levels for the estimated state values. In the presence of an uninformative prior, the information functions were found to perform virtually identically under all three measures of performance, F_r, F_c, and F_{fa}. The IS strategy was found to outperform all other search strategies in classification performance (F_r and F_c) at all times steps, but was significantly outperformed by the AS, TS, and LLRS strategies with respect to the false-alarm rate, F_{fa}. Therefore, it can be concluded that for the Bernoulli sensor model the IS strategy with the mutual-information function, $\hat{\varphi}_I$, leads to the best classification performance. The AS, TS, and LLRS strategies exhibit similar performance, and lead to the lowest rate of false alarms in all of the Bernoulli simulations.

3.6.2 Poisson Sensor Model

The rate of correct classification and the rate of false alarms for the Poisson sensor model with informative prior are plotted in Figs. 3.7 and 3.8, respectively, for different information functions. These results show that for $0 \leq k \leq 180$ the functions $\hat{\varphi}_I$ and $\hat{\varphi}_{R_2}$ perform similarly, and for $k > 180$ the function $\hat{\varphi}_{R_2}$ achieves the best performance with respect to both classification, and false alarms. The function $\hat{\varphi}_{R_2}$ also

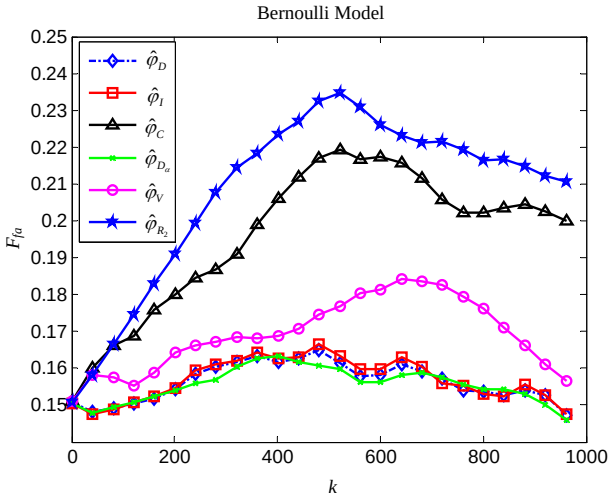


FIGURE 3.5: False-alarm rate with informative prior.

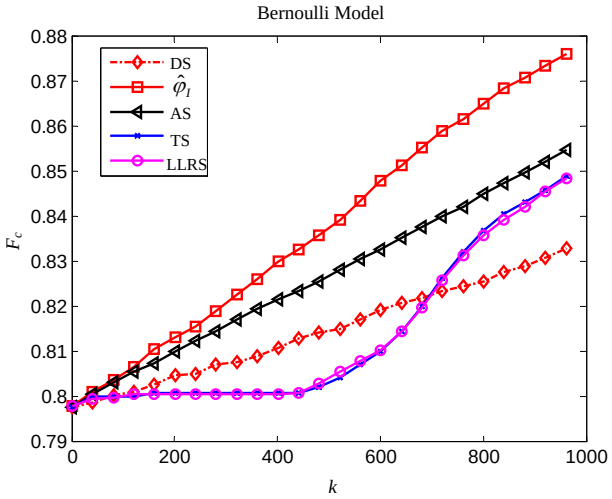


FIGURE 3.6: Correct-classification rate with informative prior.

achieves the highest value of F_r among all information functions (results are omitted for brevity). Therefore, for the Poisson sensor model, the IS strategy is compared to the DS, AS, TS, and LLRS strategies using $\hat{\varphi}_{R_2}$, as shown in Fig. 3.9. It can be seen from Fig. 3.9 that the IS strategy achieves the highest rate of correct classification for high-risk state targets. The simulations show that, for the Poisson sensor model, the IS strategy outperforms the other search strategies also with respect to the rates

of correct classification and false alarms.

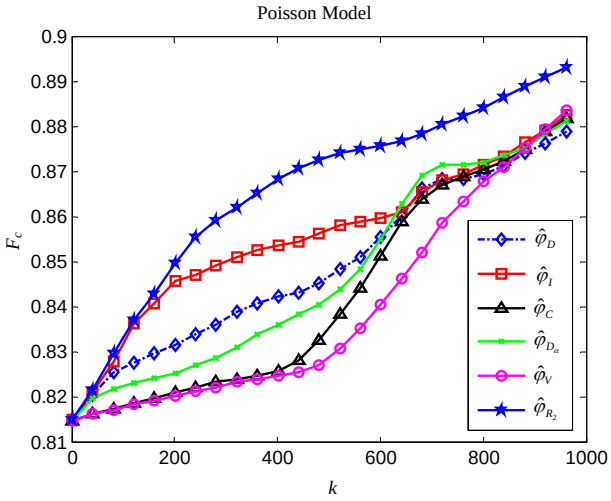


FIGURE 3.7: Correct-classification rate with informative prior.

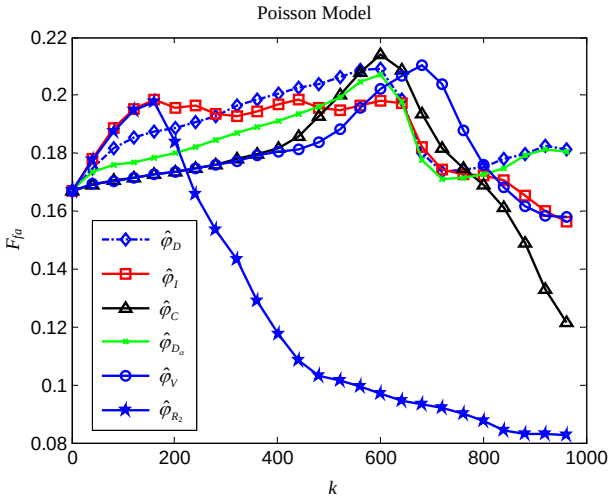


FIGURE 3.8: False-alarm rate with informative prior.

In the presence of an informative prior with large noise, all information functions perform similarly with respect to F_c. However, $\hat{\varphi}_{R_2}$ achieves slightly better performance with respect to F_r, and significantly better performance with respect to the rate of false alarms, shown in Fig. 3.10. When compared to other search strategies, IS leads to significantly higher rates of correct classification, F_c (Fig. 3.11) and F_r,

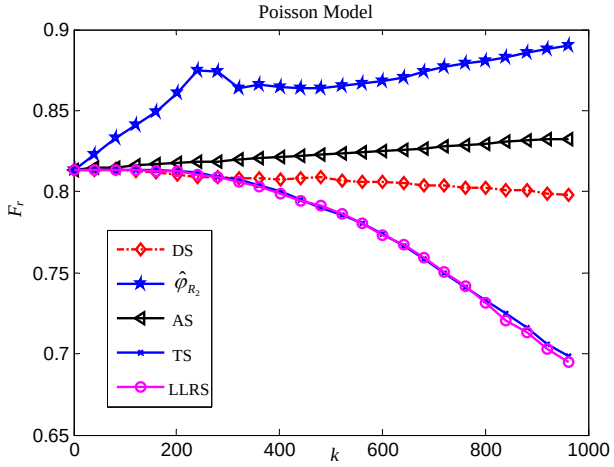


FIGURE 3.9: Correct-classification rate for high-risk state targets, with informative prior.

and to a significantly lower rate of false alarms, F_{fa} (Fig. 3.12). Therefore, for the Poisson sensor model, the IS strategy with quadratic entropy, $\hat{\varphi}_{R_2}$, leads to the best sensor performance overall, in the presence of both an informative prior and an informative prior with large noise.

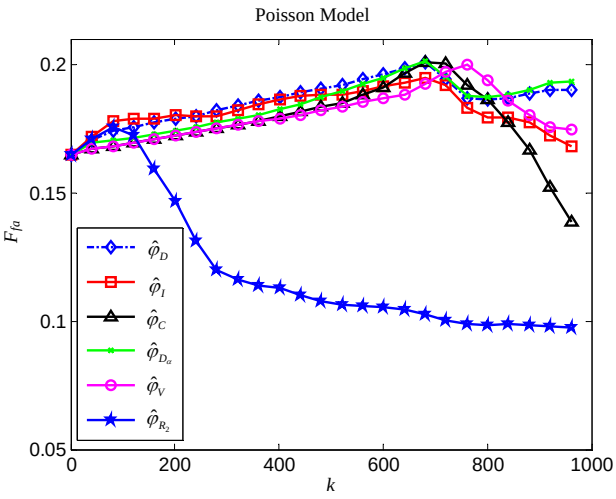


FIGURE 3.10: False-alarm rate with large-noise informative prior.

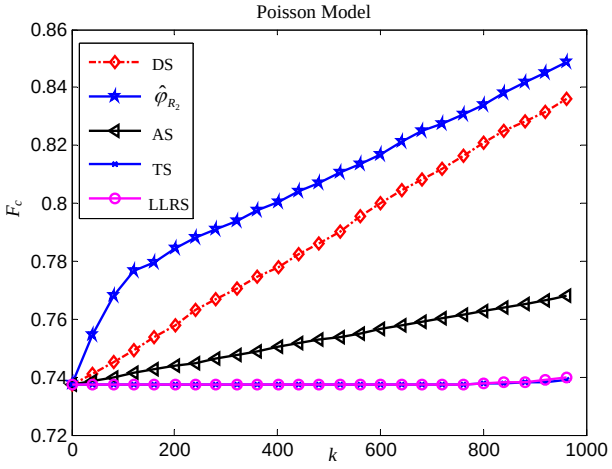


FIGURE 3.11: Correct-classification rate with large-noise informative prior.

3.6.3 Binomial Sensor Model

The same set of simulations performed for the Bernoulli and Poisson sensor models were performed using the binomial sensor model. The results (omitted here for brevity) showed that, for the binomial model, the IS strategy with the quadratic-entropy function $\hat{\varphi}_{R_2}$, achieves the best performance of all strategies with respect to all three performance measures, F_c, F_r, and F_{fa}. Subsequently, this IS strategy was compared to DS, AS, TS, and LLRS, using an infinite time horizon, defined as $f \to \infty$. In practice, an infinite-time horizon is simulated by letting $f >> c$, such that $F_c \to 1$, $F_r \to 1$, and $F_{fa} \to 0$, as $k \to f$. For this study, the number of cells in $\mathcal{K}$ is chosen as $c = 50$, and measurements are obtained up to a final time $f = 3000$.

As shown by the results in Figs. 3.13 and 3.14, in the limit of $k \to f$, the IS strategy classifies all cells correctly, and eliminates all false alarms. It can be seen from Figs. 3.13-3.14 that the AS and DS strategies approach the IS performance with a slower rate of convergence, and leave a small percentage of cells improperly classified. Also, although the TS and LLRS strategies have the advantage of displaying low false-alarm rates (Section 3.6.1), by always selecting cells with a high expected

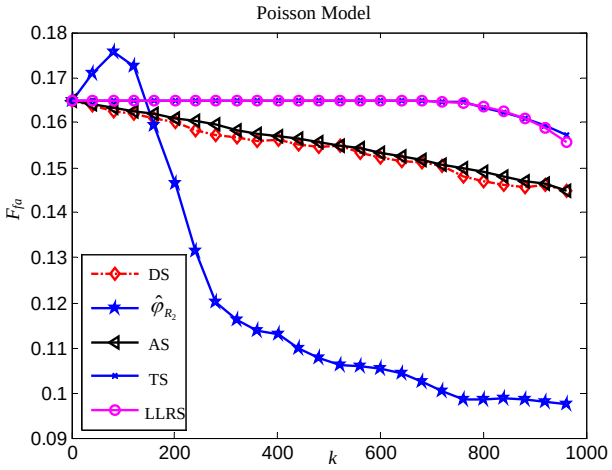


FIGURE 3.12: False-alarm rate with large-noise informative prior.

confidence level, TS and LLRS also leave a substantial percentage of cells improperly classified even as $k \to f$ (Figs. 3.13-3.14). Therefore, it can be concluded that the IS strategy is better suited to infinite-horizon problems in which, given sufficient time and sensor measurements, it is desirable to correctly classify all cells, or all targets.

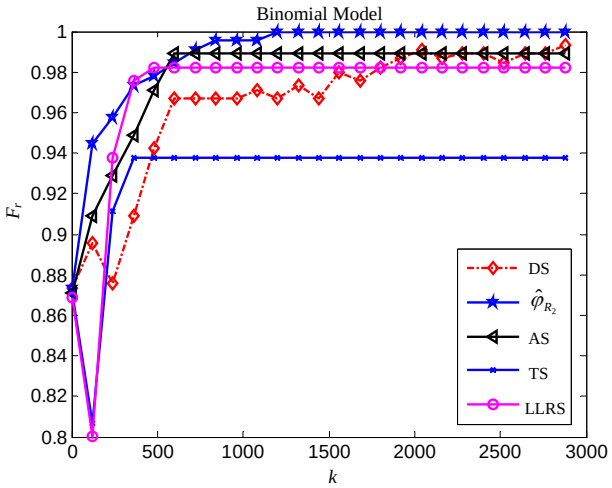


FIGURE 3.13: Correct-classification rate for high-risk state targets, with informative prior, $c = 50$, and $f = 3000$.

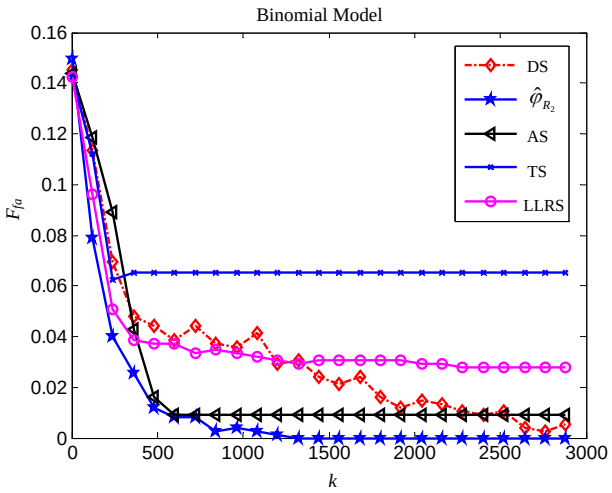


FIGURE 3.14: False-alarm rate with informative prior, $c = 50$, and $f = 3000$.

3.6.4 Mixture-of-Binomials Sensor Model

The rates of correct classification, F_c and F_{fa}, for the mixture-of-binomials sensor model are plotted in Figs. 3.15-3.16, using different information functions, and an informative prior. The results in Fig. 3.15 show that $\hat{\varphi}_C$ achieves the highest values of F_r for $0 \leq k \leq 375$, but is outperformed by $\hat{\varphi}_{R_2}$ for $k > 375$. Although none of the information functions clearly outperforms the others with respect to F_r, $\hat{\varphi}_{R_2}$ outperforms all functions, and at all time steps, with respect to F_{fa} (Fig. 3.16) and F_c (results not shown for brevity). Therefore, the IS strategy is implemented using $\hat{\varphi}_{R_2}$, and compared to DS, AS, TS, and LLRS strategies in Figs. 3.17-3.18. It can seen from Figs. 3.17-3.18, and from the F_c comparison (omitted for brevity), that, for the mixture-of-binomials sensor model and an informative prior, the IS strategy outperforms all other strategies, both with respect to classification and false alarms.

Extensive mixture-of-binomials sensor simulations were also conducted using the uninformative prior and the informative prior with large noise. It was found that the information functions all perform similarly with respect to F_c, F_r, and F_{fa}. Also, while the IS strategy led to significantly higher values of F_r compared to the other

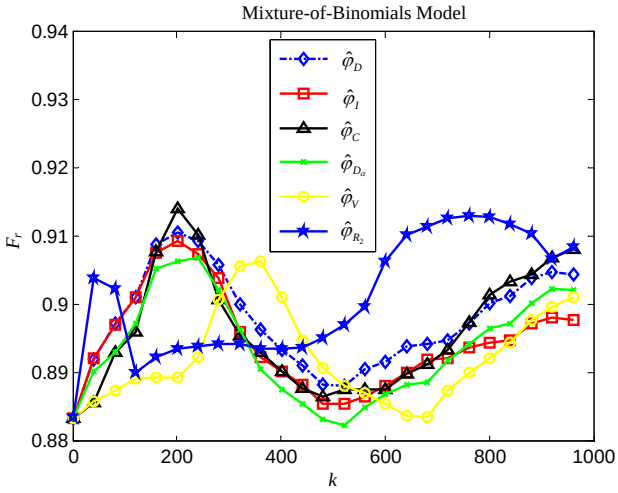


FIGURE 3.15: Correct-classification rate for high-risk state targets, with informative prior.

search strategies, IS was outperformed by the DS strategy with respect to F_c and F_{fa}.

3.6.5 Classical Sensor Model

The classification performance (F_c) of the six information functions considered in this chapter was found to be approximately equal for all three levels of sensor noise ($\sigma_i = 1, 3, 5$), and for all priors $p(X_i)$. However, as the noise level increases from $\sigma_i = 1$ to 5, the information functions increasingly differ in their rates of false alarms, F_{fa}, and in their rates of correct high-risk-target classification, F_r. In particular, $\hat{\varphi}_I$ and $\hat{\varphi}_D$ achieve slightly better values of F_r than other functions for $\sigma_i = 1, 3$ and $\sigma_i = 5$, respectively, but also display the highest rates of false alarms. $\hat{\varphi}_{R_2}$ achieves slightly lower values of F_r than $\hat{\varphi}_I$ and $\hat{\varphi}_D$, but displays the lowest rate of false alarms of all information functions, for all values of σ_i.

Therefore, for the classical sensor model, the IS strategy is implemented using $\hat{\varphi}_{R_2}$, and is compared to the DS, AS, TS, and LLRS strategies in Figs. 3.19-3.21. It can be seen that, in the presence of low noise, the IS strategy achieves the highest

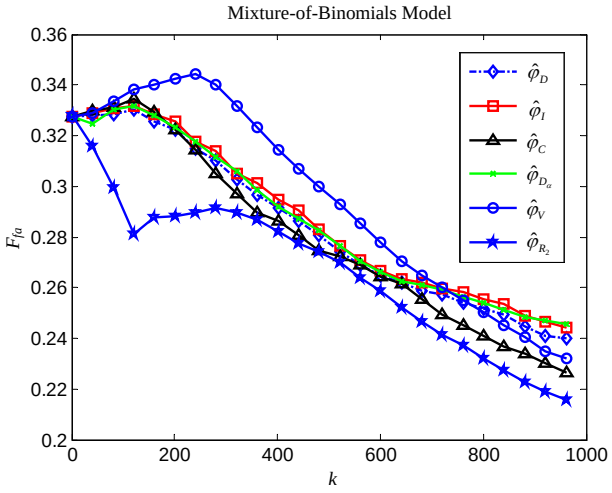


FIGURE 3.16: False-alarm rate with informative prior.

correct-classification rate for high-risk targets (Fig. 3.19), as well as for all cells (results omitted for brevity), and the lowest false-alarm rate (Fig. 3.20). As the noise level in the sensor model is increased, the IS strategy outperforms the other strategies even more significantly with respect to F_c and F_r (Fig. 3.21), but is then outperformed by TS and LLRS with respect to F_{fa}. It can be concluded that, for the classical sensor model, the IS strategy leads to the best classification performance but, in the presence of high noise, the TS and LLRS strategies lead to lower rates of false alarms.

3.7 Summary

Many sensing tasks can be formulated as estimation problems involving partial or imperfect measurements. Therefore, the utility of the sensor decisions can be represented by the information value associated with the targets and sensor measurements. Information theoretic functions are a natural choice for representing the value of information but, typically, require knowledge of the belief state before and after the measurements arrive. This chapter has presented a general and systematic approach

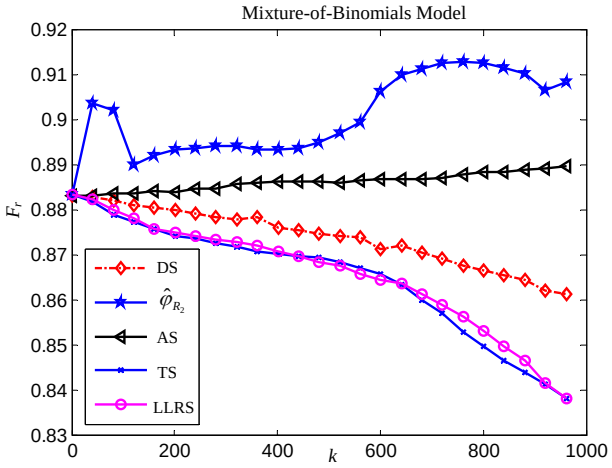


FIGURE 3.17: Correct-classification rate for high-risk state targets, with informative prior.

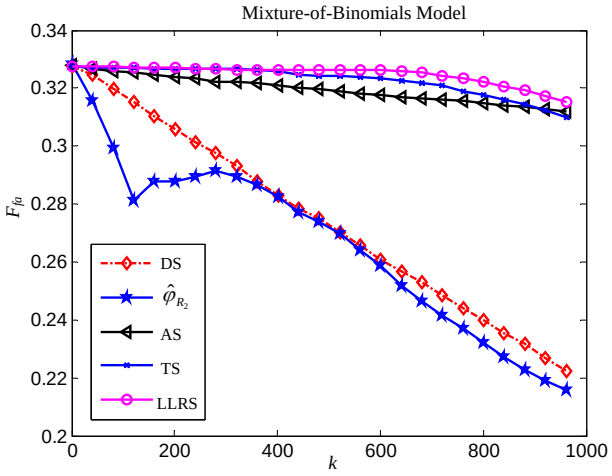


FIGURE 3.18: False-alarm rate with informative prior.

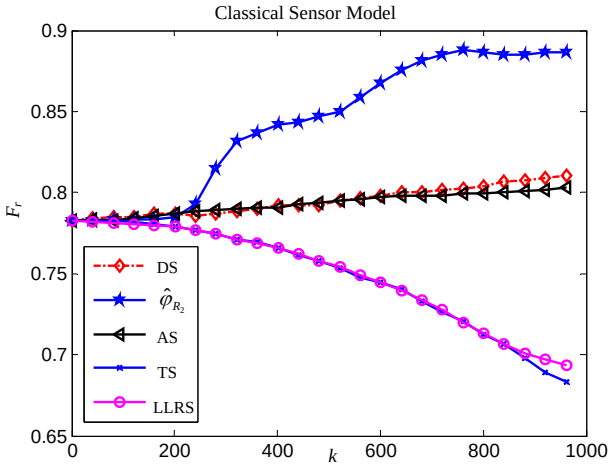


FIGURE 3.19: Correct-classification rate for high-risk targets, with informative prior and $\sigma_i = 1$.

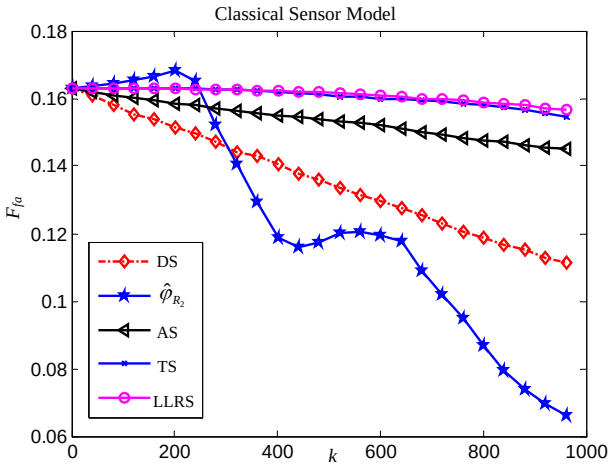


FIGURE 3.20: False-alarm rate with informative prior, and $\sigma_i = 1$.

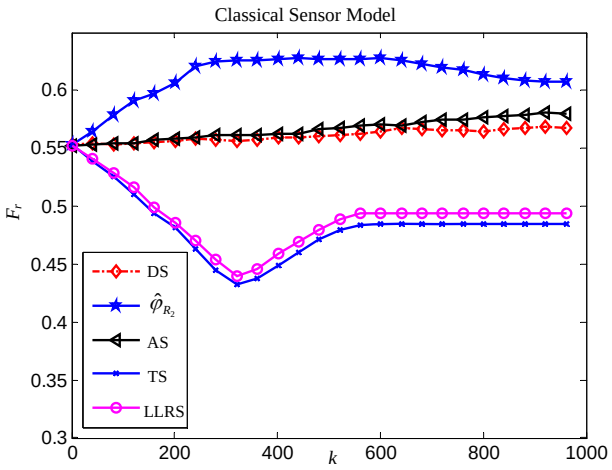


FIGURE 3.21: Correct-classification rate for high-risk targets, with informative prior and $\sigma_i = 5$.

for deriving information value functions that represent the expected utility of the sensor decisions in a canonical sensor planning problem. The resulting information functions can then be implemented by an information-driven search (IS) strategy that selects the measurement sequence with the highest expected information value. The IS strategy is compared to other search strategies previously presented in the literature by means of extensive numerical simulations involving several sensor models, priors, and decision rules. Consequently, the comparative performance results can be generalized beyond a particular sensor type or application. It was found that quadratic entropy leads to the most effective information function, and that the corresponding IS strategy outperforms all others with respect to classification performance. In the presence of prior information, IS also displays the lowest rate of false alarms. However, when prior information is absent or very noisy, the task-driven and log-likelihood-ratio strategies achieve the lowest false-alarm rates for Bernoulli, mixture-of-binomials, and classical sensor models.

4

Information Roadmap Method for Robotic Sensor Path Planning

PRM algorithms have been shown very effective at planning a collision-free path for robots with many degrees of freedom for the purpose of moving from an initial to a final configuration in a workspace by a minimum distance [38–41]. Along a separate line of research, several information-driven sensor planning algorithms have been developed to plan a sensor measurement process based on the expected value of information of the measurements. The information roadmap method (IRM) presented in this section combines these two lines of research, by using a value-of-information metric to generate a new hybrid sampling strategy that increases the milestone density near targets with high information value, while also covering wide-open regions and narrow passages. As a result, IRM presents several advantages over existing sensor path planning approaches, such as, coverage path-planning [3, 37], random [3], grid [35], and probabilistic search strategies [2,35,36,61,79]. By extending the concept of a roadmap to the sensor planning problem, IRM can account for the geometries of the targets and of a moving sensor's field of view, and can be applied to robotic sen-

sors with a finite platform geometry, moving in obstacle-populated workspaces. Also, thanks to the information-value metric presented in Chapter 3, IRM accounts for the influence of operating and environmental conditions on the measurements that can be obtained along the path, prior to determining the features or classification of the targets that can be measured by following it.

In this chapter we assume that only one robotic sensor is deployed in the workspace. The purpose for deploying the robotic sensor in $\mathcal{W}$ is to obtain additional measurements to properly classify a subset of targets in T. It is assumed that to each target $\mathcal{T}_i \in T$ there is associated a discrete and, possibly, random variable, y_i, with finite range $\mathcal{Y} = \{y_i^1, \ldots, y_i^p\}$, representing its classification. Due to limited sensor measurements or to targets being buried underground, y_i is non-observable or *hidden*, and must be inferred from a set of measurements, $M_i = \{m_{i1}, \ldots, m_{if}\}$. Every measurement $m_{i\ell} \in M_i$ also is a discrete and random variable, with finite range $\mathcal{M}_{i\ell} = \{m_{i\ell}^1, \ldots, m_{i\ell}^{N_\ell}\}$, where $m_{i\ell}^k$ denotes the k^{th} value of $m_{i\ell}$. After the robotic sensor obtains the set of measurements M_i from $\mathcal{T}_i$, y_i can be inferred from an available sensor model, which is typically given by a known joint probability mass function, $P(y_i, M_i)$.

While the platform $\mathcal{A}$ must avoid collisions with the obstacles B, in order to obtain the measurements M_i from a target $\mathcal{T}_i$ the sensor's field of view $\mathcal{S}$ must intersect $\mathcal{T}_i$. Since $\mathcal{S}$ is mounted on $\mathcal{A}$, the platform motion must be planned in concert with the sensor measurements, and the path τ must simultaneously avoid obstacles while searching for targets. Let the measurement set of a robotic sensor along a path τ be defined as $M(\tau) = \{M_i \mid \mathcal{T}_i \cap \mathcal{S}(\mathbf{q}) \neq \emptyset,\ \tau(s) = \mathbf{q},\ s \in [0,1],\ i \in I_T\}$, where $\mathcal{S}(\mathbf{q})$ is the subset of $\mathcal{W}$ occupied by $\mathcal{S}$ at a configuration $\mathbf{q}$, along τ. Then, the robotic sensor path τ between $\mathbf{q}_0$ and $\mathbf{q}_f$ must achieve multiple, simultaneous objectives:

1. Avoid all obstacles in $\mathcal{W}$

2. Minimize the distance traveled from $\mathbf{q}_0$ to $\mathbf{q}_f$

3. Maximize the information value of $M(\tau)$

In order to meet objectives (2)-(3), the robotic sensor performance is defined by an additive reward function that represents the profit of the information obtained along the path τ:

$$R(\tau) = w_V \cdot V(\tau) - w_D \cdot D(\tau) \tag{4.1}$$

Where, $V(\tau)$ is the information value of $M(\tau)$, and $D(\tau)$ is the distance traveled along τ, as defined in Section 4.3. Then, a robotic sensor path that meets objectives (1)-(3) is obtained by solving the following problem:

Problem 1 (Sensor Path-Planning Problem)**.** *Given a layout $\mathcal{W}$ and a joint probability mass function, $P(y_i, M_i)$, $\forall i \in I_T$, find a path τ^* for a robotic sensor with platform $\mathcal{A}$ and field-of-view $\mathcal{S}$ that connects $\mathbf{q}_0$ to $\mathbf{q}_f$ in $\mathcal{C}_{free}$, and maximizes the profit of information in (4.1).*

4.1 Measurement Information Value

A basic difficulty in sensor planning consists of assessing the value or utility of the sensor measurements prior to obtaining them from the targets. Several information-theoretic metrics have been proposed for this purpose in Chapter 3. Cross entropy was used in [20] to solve a multisensor-multitarget assignment problem, and in [21,22] to manage agile sensors with Gaussian models for target detection and classification. Entropy and the Mahalanobis distance measure were used in [18] for sensor selection in *ad-hoc* sensor networks. In this chapter, the expected entropy reduction (EER) described in Chapter 3 is used to represent the value of information, V, in terms of the joint PMF $P(y_i, M_i)$, known from the sensor model (Section 2.2). Other

information-theoretic functions can be similarly applied, as shown in Section 4.2, to construct information roadmaps for other sensor applications, e.g., [18, 20, 22].

Recall that the entropy reduction is formulated in terms of conditional entropy, which can be used to represent the uncertainty in a discrete and random variable Y, given the value of another discrete and random variable Z, based on their joint and conditional PMFs,

$$H(Y|Z) = -\sum_{z}\sum_{y} P(y,z)\log_2 P(y|z) \tag{4.2}$$

where, $\sum_y$ denotes marginalization over the range of Y [63]. Although information entropy is not additive, it can be shown [80] that the entropy reduction

$$\Delta\hat{H}(Y;\ Z_j \mid Z_i) \equiv H(Y \mid Z_i) - H(Y \mid Z_i, Z_j) \tag{4.3}$$

is additive, and represents the reduction in uncertainty brought about by Z_j, given prior information or evidence about Z_i.

Since sensor measurements from a target $\mathcal{T}_i$ are sought to reduce the uncertainty in the classification variable y_i, entropy reduction is used here to represent the value of a new (posterior) set of measurements M_i, given an *a-priori evidence set* $\mathcal{E}_i^0$, which may include known environmental conditions, as well as the measurements and mode of a previously-deployed sensor. The superscript $(\cdot)^0$ denotes a set of random variables whose values are known *a priori*. Since the *actual* entropy reduction, $\Delta\hat{H}(y_i;\ M_i \mid \mathcal{E}_i^0)$, cannot be determined prior to measuring M_i, the *expected* entropy reduction (EER), defined as,

$$\Delta H(y_i;\ M_i \mid \mathcal{E}_i^0) \equiv H(y_i \mid \mathcal{E}_i^0) - \sum_{\ell=1}^{f}\sum_{k=1}^{N_\ell}\left[H(y_i \mid m_{i\ell} = m_{i\ell}^k)P(m_{i\ell} = m_{i\ell}^k \mid \mathcal{E}_i^0)\right], \quad \forall i \tag{4.4}$$

is used to represent the expected reduction in uncertainty in y_i that would be brought about by M_i, given $\mathcal{E}_i^0$.

In (4.4), the conditional entropy $H(y_i \mid \mathcal{E}_i^0)$ is computed from the definition, and from the posterior PMF $P(y_i \mid \mathcal{E}_i^0)$ computed by a junction-tree BN inference algorithm [67]. The conditional entropy $H(y_i \mid m_{i\ell} = m_{i\ell}^k)$ is computed by using the posterior PMF,

$$\begin{aligned} P(y_i \mid m_{i\ell}^k) &= \frac{P(y_i)P(m_{i\ell}^k \mid y_i)}{P(m_{i\ell}^k)} \\ &= \frac{P(y_i)\sum_j \sum_l P(m_{i\ell}^k \mid f_{ij} = f_{ij}^l)P(f_{ij} = f_{ij}^l \mid y_i)}{\sum_j \sum_l P(m_{i\ell}^k \mid f_{ij} = f_{ij}^l)\sum_\jmath P(f_{ij} = f_{ij}^l \mid y_i = y_i^\jmath)P(y_i = y_i^\jmath)}, \quad \forall \ell, k \end{aligned} \tag{4.5}$$

where $\sum_j$ and $\sum_l$ denote marginalization over the range of all features $f_{ij} \in F_i$, and $\sum_\jmath$ denotes marginalization over the range of y_i (i.e., $\mathcal{Y}$). Now, all of the probabilities in (4.5) are known from the BN CPTs $P(y_i)$, $P(F_i \mid y_i)$, and $P(M_i \mid S_i, E_i, F_i)$. Because, the latter can be used to compute $P(M_i \mid F_i)$, either by marginalization or by using the evidence $\mathcal{E}_i^0$, when available. Equation (4.5) is derived using the simplification,

$$P(M_i \mid y_i) = \sum_j \sum_l P(M_i \mid f_{ij}^l, y_i)P(f_{ij}^l, y_i) = \sum_j \sum_l P(M_i \mid f_{ij}^l)P(f_{ij}^l, y_i), \quad \forall i \tag{4.6}$$

obtained by noting that y_i and M_i are d-separated given F_i, because of the serial connection between the respective nodes (Fig. 2.1). Finally, the last term in (4.4) is computed as $P(M_i \mid \mathcal{E}_i^0) = \sum_j \sum_l P(M_i \mid f_{ij} = f_{ij}^l)P(f_{ij} = f_{ij}^l \mid \mathcal{E}_i^0)$, where $P(F_i \mid \mathcal{E}_i^0)$ is obtained using a junction-tree BN inference algorithm [67].

In the presence of multiple targets, the value of a set of measurements $M(\tau)$ along the path τ is the cumulative EER,

$$V(\tau) = V[M(\tau)] \equiv \sum_{M_i \in M(\tau)} \Delta H(y_i;\ M_i \mid \mathcal{E}_i^0) \tag{4.7}$$

where, y_i, M_i, and $\mathcal{E}_i^0$ are the classification variable, measurements, and *a-priori* evidence set corresponding to target $\mathcal{T}_i$, respectively, and $M(\tau)$ is defined at the beginning of this chapter. Since the BN sensor model holds for any target in $\mathcal{W}$, every term in the summation in (4.7) can be computed from the corresponding *a-priori* evidence set $\mathcal{E}_i^0$ using eqs. (4.2)-(4.6). The methodology presented in the next section presents a new sampling strategy that is based on a probability density function obtained from the EER in (4.4). By combining this information-based sampling distribution with π_H in (2.12), a roadmap is constructed that captures the information value of the targets as well as the connectivity of $\mathcal{C}_{free}$.

4.2 Learning Phase: Analysis of the Sampling Distribution

As shown in Chapter 3, various information metrics can be used to estimate the expected information value of the measurements from a target $\mathcal{T}_i$, prior to obtaining the measurements, and prior to determining the target's features and classification. In this chapter we choose EER as the information metric. In order for a sensor with a bounded field of view $\mathcal{S}$ to obtain the measurements M_i from the i^{th} target, $\mathcal{S}$ must intersect the target geometry $\mathcal{T}_i$. Therefore, the subsets of $\mathcal{W}$ where the sensor can make target measurements can be defined similarly to C-obstacles, as follows:

Definition 1 (C-Target)**.** *The target $\mathcal{T}_i$ in $\mathcal{W}$ maps in the robot's configuration space, $\mathcal{C}_{free}$, to the C-target region $\mathcal{CT}_i = \{\mathbf{q} \in \mathcal{C}_{free} \mid \mathcal{S}(\mathbf{q}) \cap \mathcal{T}_i \neq \varnothing\}$.*

The union of all the C-targets in $\mathcal{C}_{free}$ is the C-target region $\mathcal{CT}$, and the union of all C-targets corresponding to a robot configuration $\mathbf{q}$ is the set of C-targets $\mathcal{CT}(\mathbf{q}) = \{\mathcal{CT}_i \mid i \in I_T,\ \mathcal{S}(\mathbf{q}) \cap \mathcal{T}_i \neq \varnothing\}$. The measurement set that can be obtained in $\mathbf{q}$ is the set of measurements corresponding to these C-targets, i.e., $M(\mathbf{q}) = \{M_i \mid i \in I_T,\ \mathcal{CT}_i \in \mathcal{CT}(\mathbf{q})\}$. Therefore, the information value of the measurements that can be obtained in a robotic sensor configuration $\mathbf{q} \in \mathcal{C}_{free}$ is the

cumulative EER,

$$V(\mathbf{q}) = V[M(\mathbf{q})] \equiv \sum_{M_i \in M(\mathbf{q})} \Delta H(y_i;\ M_i \mid \mathcal{E}_i^0) \tag{4.8}$$

It follows that a high density of milestones with high information value can be obtained by sampling the probability density function defined as,

$$\pi_V(\mathbf{q}) = \frac{V(\mathbf{q})}{\int_{\mathcal{CT}} V(\mathbf{q}) d\mathbf{q}} \tag{4.9}$$

which is proportional to the EER function $V(\mathbf{q})$, and is normalized by the total EER of the C-target region.

The above PDF is combined with a uniform PDF π_U and a Gaussian PDF π_G obtained by the bridge test in order to construct a roadmap that also captures the connectivity of $\mathcal{C}_{free}$ by covering narrow passages and wide-open regions. A new hybrid strategy is obtained from the sampling strategy, (2.12),

$$\begin{aligned} \pi &= v_2 \pi_V + (1 - v_2)\pi_H \\ &= v_2 \pi_V + v_1(1 - v_2)\pi_G + (1 - v_1)(1 - v_2)\pi_U \end{aligned} \tag{4.10}$$

where, $0 \leq v_1 \leq 1$ and $0 \leq v_2 \leq 1$ are user-defined parameters that are chosen to emphasize the narrow passages versus wide-open areas, and to emphasize information value versus connectivity, respectively. In the actual implementation, the PDF π is discretized into a PMF over $\mathcal{W} \in \Re^2$, as shown in Appendix B. The sampling distribution π obtained by the algorithm in Appendix B is plotted in Fig. 4.1 for a simple example involving a robotic sensor that can translate freely but cannot rotate, and has a square platform geometry $\mathcal{A}$ (grey), and a larger square filed-of-view $\mathcal{S}$ (white). This robotic sensor navigates the workspace shown in Fig. 4.1.a, where the obstacles' geometry are shown in black, and the target geometries are shown in color patterns that are representative of their information value. The sampling

distribution over $\mathcal{C}$, which in this simple example is equal to $\mathcal{W}$ because $\mathbf{q} = [x\ y]^T$, is plotted in Fig. 4.1.b, where C-obstacles (black) have zero sampling probability, and C-targets have sampling probability proportional to their information value (EER). As shown in Sections 4.4-4.5, the same approach is applicable to robotic sensors with more degrees of freedom, such as robots that are capable of rotating, for which $\mathcal{C} \subset \Re^3$, as well as to any other geometries for $\mathcal{W}$, $\mathcal{A}$, and $\mathcal{S}$.

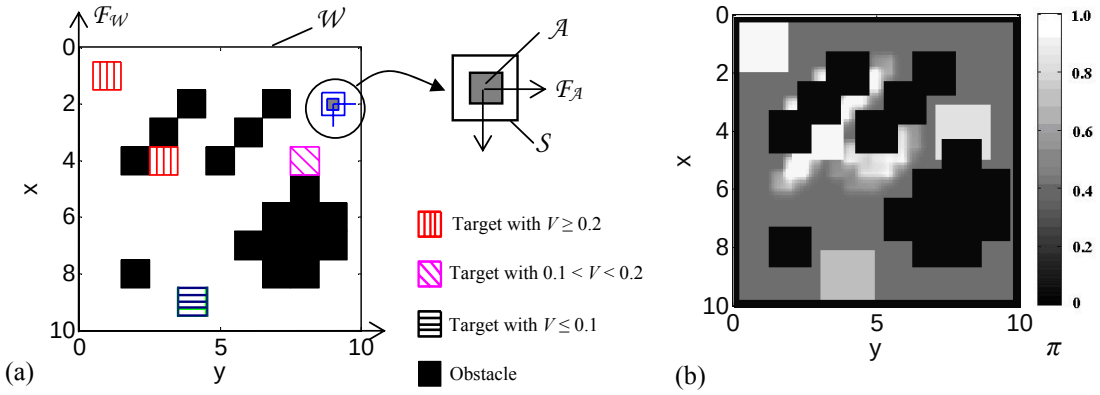


FIGURE 4.1: Example of sampling probability distribution π (b) for the non-rotating robotic sensor with platform $\mathcal{A}$ and field of view $\mathcal{S}$, navigating the workspace $\mathcal{W}$ in (a).

After the sampling distribution π is computed by the algorithm in Appendix B, it is sampled N_m times to obtain the set of milestones L that constitute the nodes of the roadmap $G = (L, A)$. Subsequently, the set of arcs A is obtained by a local planner that connects every milestone $c_i \in L$ to every other milestones with a straight line segment. The lengths of these straight-line segments are used to sort the milestones, such that only the k nearest milestones to c_i are elected as its candidate neighbors. Where, k is a used-defined parameter that is chosen based on N_m and on the complexity of $\mathcal{W}$. For each candidate neighbor of c_i, the collision-check algorithm presented in [41] is implemented to check whether a simple path between them (typically a straight line segment) is collision free, by discretizing the

path into a sequence of configurations. Then, the candidate neighbors with collision-free paths are connected to c_i to construct G. After G is completed, it can be used to find the path with maximum information profit (4.1) between any pair of $\mathbf{q}_0$ and $\mathbf{q}_f$, as explained in the next section.

4.3 Information-driven Query Phase

After the informational roadmap G is constructed by the method presented in the previous section, the query phase seeks a path τ^* from $\mathbf{q}_0$ to $\mathbf{q}_f$ of maximum information profit (Problem 1). A query consists of connecting $\mathbf{q}_0$ and $\mathbf{q}_f$ to G by searching for two milestones $c_0, c_f \in G$ that have the shortest collision-free distance to $\mathbf{q}_0$ and $\mathbf{q}_f$, respectively, and then finding the path from c_0 to c_f of maximum information profit. A path from c_0 to c_f in G is a sequence of adjacent cells that are pairwise connected by a simple collision-free path represented by the arc between them, e.g., $\tau = \{c_0, c_i, c_j, \ldots, c_f\}$, with $(c_i, c_j) \in A \subset G$, for any consecutive pair of indices in the index set I_τ of τ. Since every milestone represents a robot configuration, an additive distance metric can be defined for every pair $(c_i, c_j) \in A$, and used to compute the total path distance $D(\tau)$ in (4.1). A comparative study of distance metrics for PRM algorithms was performed in [39], indicating that the scaled Euclidian metric is generally a good choice in terms of performance and computational efficiency. Therefore, in this chapter, a scaled Euclidian distance is adopted that changes the relative importance of position and orientation components of $\mathbf{q}$ through a set of weights organized in a diagonal and positive-definite matrix $W \in \Re^{3\times 3}$. Then, the total path distance is given by the sum of all weighted Euclidian norms along τ, i.e.,

$$D(\tau) = \sum_{i=0}^{f-1} \|W \cdot [\mathbf{q}(c_{i+1}) - \mathbf{q}(c_i)]\| \tag{4.11}$$

where, $\mathbf{q}(c_i)$ denotes the robot configuration represented by c_i, $(\cdot)$ represents matrix multiplication, and $\|\cdot\|$ represents the Euclidian norm [19].

As shown in [39], planners based on the A^* algorithm are the most effective at searching for the path of minimum total distance in G. The A^* algorithm explores G iteratively, starting at c_0 and visiting every neighbor node c_i, to which a cost function is assigned by estimating the minimum-cost path from c_0 to c_f, through c_i [81, 82]. Based on the principle of optimality [83], this cost can be estimated as the sum of the actual cost of a path from c_0 to c_i, $g(c_i)$, plus an estimate of the minimum-cost path from c_i to c_f obtained from a heuristic function, $h(c_i)$. If the heuristic function is chosen such that $h(c_i)$ is always less than the actual cost of the (same) path from c_i to c_f, then the A^* algorithm is guaranteed to return a path of minimum cost whenever such a path exists in G [81, 82]. After a node is visited, the algorithm stores only the path of minimum cost, and labels the node as visited, assigning it a pointer to its parent node. This process forms a spanning tree T_s of the subset of G that has already been explored, and brings about considerable computational savings compared to other graph-searching algorithms [14, 50].

Since the goal of the robotic-sensor path is to maximize the measurement information profit, after connecting $\mathbf{q}_0$ and $\mathbf{q}_f$ to G, the information-driven query phase presented in this chapter seeks the path τ^* with the maximum value of $R(\tau)$ in (4.1), with $V(\tau)$ defined in (4.7), and $D(\tau)$ defined in (4.11). Thus, the A^* algorithm is applied by defining the actual cost of a path $\tau^s_{0,i}$ connecting c_0 to c_i in the current spanning tree T_s as

$$g(c_i) \equiv -R(\tau^s_{0,i}) = -w_V \cdot V(\tau^s_{0,i}) + w_D \cdot D(\tau^s_{0,i}). \tag{4.12}$$

In order to guarantee that the cost estimated by the heuristic function is less than the actual cost, $h(c_i)$ is based on the cumulative EER of all measurements remaining after $\tau^s_{0,i}$, $\bar{M}(\tau^s_{0,i})$, and on the length of a straight line segment $s_{i,f}$ connecting $\mathbf{q}(c_i)$

to $\mathbf{q}(c_f)$ in $\mathcal{C}$:

$$h(c_i) = -\sum_{\bar{M}(\tau^s_{0,i})} w_V \cdot \Delta H(y_i;\ M_i \mid \mathcal{E}^0_i) + w_D \cdot D(s_{i,f}) \tag{4.13}$$

Where, the complement measurement set of a path is defined as $\bar{M}(\tau^s_{0,i}) = \{M_i \mid i \in I_T,\ M_i \notin M(\tau^s_{0,i})\}$. The estimated total cost for a path from c_0 to c_f through c_i is $f(c_i) = g(c_i) + h(c_i)$. Thus, it includes the (negative) cumulative EER of all targets in $\mathcal{W}$. The actual set of measurements $M(\tau_{i,f})$ obtained along a path $\tau_{i,f}$ from c_i to c_f in G is a subset of $\bar{M}(\tau^s_{0,i})$, because the EER is conditioned upon the evidence $\mathcal{E}^0_i$, thereby preventing repetition of the same set of measurements M_i. The inequality,

$$\sum_{\bar{M}(\tau^s_{0,i})} \Delta H(y_i;\ M_i \mid \mathcal{E}^0_i) > \sum_{M(\tau_{i,f})} \Delta H(y_i;\ M_i \mid \mathcal{E}^0_i) \tag{4.14}$$

then holds for any path $\tau_{i,f}$ that is concatenated with $\tau^s_{0,i}$ to connect c_0 to c_f through c_i, because $\Delta H(\cdot) \geq 0$ by the properties of mutual information [63]. Since the length of $s_{i,f}$ is always less than or equal to the shortest path between c_i and c_f in G, it also follows that $h(c_i) < -R(\tau_{i,f})$. Therefore, the A^*-type search shown in Appendix C always returns the path τ^* of minimum cost or maximum information profit in G.

4.4 Information Roadmap Implementation for Robotic Demining

The IRM methodology presented in the previous section is demonstrated through a demining system application, in which the path of a robot with an on-board ground-penetrating radar (GPR) is planned based on prior infrared (IR) sensor measurements and environmental information available from a minefield $\mathcal{W} \subset \Re^2$. The purpose for deploying a GPR robotic sensor in $\mathcal{W} \subset \Re^2$ is to infer the classification of targets that are either mines or clutter objects, and are buried in heterogeneous soils, under non-uniform environmental conditions [84]. The classification variable

of every target, y_i, is non-observable and has two possible values $\mathcal{Y} = \{y_i^1,\ y_i^2\}$. y_i can be inferred from a set of measurements $M_i = \{d_{m_i}, z_{m_i}, s_{m_i}\}$ that pertain to the target features, namely, depth (d_i), size (z_i), and shape (s_i). The measured target features are typically extracted from raw sensor measurements through signal processing techniques, and thus contain random measurement errors and errors caused by unfavorable environmental conditions [85]. The actual target features $F_i = \{d_i, z_i, s_i\}$, together with y_i, can be inferred from M_i and any prior evidence $\mathcal{E}_i^0$ using a GPR sensor model (Section 4.4.1). It is assumed that an IR sensor previously deployed on an airborne platform, such as an unmanned air vehicle (UAV), is used to obtain cursory measurements that together with other geospatial data (e.g., topography, land cover, and satellite imagery) provide $\mathcal{E}_i^0$ for $\forall i \in I_T$, as well as estimated targets' and obstacles' geometries, T and B, in $\mathcal{W}$. The GPR robotic sensor, on the other hand, makes more accurate measurements on the ground, but can only visit a subset of the targets in $\mathcal{W}$ due to energy and time limitations.

The simulation of a demining sensor system developed in [1] is used to generate a rectangular minefield of chosen dimensions, or workspace, that includes several buried mines, clutter objects, obstacles, and heterogenous environmental conditions. A two-dimensional grid is superimposed on the minefield dividing it into square *bins* that are assigned a squared unit distance. Soil characteristics, vegetation, and time-varying meteorological conditions are modeled according to [86, 87], as shown in Table 4.1. The simulation assigns a set of environmental conditions to each bin, either at random or at user-specified positions. The targets are comprised of anti-tank mines (ATM), anti-personnel mines (APM), unexploded ordnance (UXO), and clutter objects (CLUT) that are sampled and reproduced using the Ordata Database [88], which contains over 5,000 explosive items and 3,000 metallic and plastic objects that resemble anti-personnel mines. Each target occupies one or more bins in the minefield depending on its size (z_i), and is characterized by a depth (d_i) and a shape

(s_i) that may take any of the values shown in Table 4.1. Thus, $\mathcal{T}_i \subset \mathcal{W}$ represents the geometry of the set of bins from which prior IR measurements are obtained for the i^{th} target detected in $\mathcal{W}$, with features z_i, d_i, and s_i. The ground robot is simulated using the nonholonomic unicycle model in $\mathcal{F}_{\mathcal{W}}$ [89, 90], with a platform geometry $\mathcal{A} \subset \mathcal{W}$ specified by the user. On-board the ground robot is a GPR sensor with a field of view geometry $\mathcal{S} \subset \mathcal{W}$, also specified by the user, that moves with the robot in $\mathcal{F}_{\mathcal{W}}$ and remains fixed with respect to $\mathcal{A}$. In the simulation, as soon as $\mathcal{S}$ intersects a bin containing a target, measurements are reproduced and deteriorated based on the target features, the sensor mode and working principles, and the environmental conditions in the bin [1, 91].

Table 4.1: Simulated minefield, sensor, and target characteristics [1].

Symbol:	**Nodes:**	**Range:**
y	Target classification	{clutter, mine, empty bin}
S	GPR mode: m_{GPR}	{depth search, resolution search, anti ground-bounce-effect search }
	IR mode: m_{IR}	{surface-mine search, shallow-buried-mine search}
E	Soil moisture (%): s_r	{dry [0, 10], wet (10, 40], saturated (> 40)}
	Soil composition: s_c	{very-sandy, sandy, high-clay, clay, silt}
	Soil uniformity: s_u	{yes, no}
	Vegetation: g	{no-vegetation, sparse, dense}
	Weather: w	{clear, overcast, raining}
	Illumination: i	{low (7-10 a.m. and 6-9 p.m.), medium (10-1 p.m.), high (1-6 p.m.)}
F	Depth (cm): d	{surface [0], shallow-buried (0, 12], buried (12, 60], deep-buried (> 60)}
	Size (cm): z	{small (2, 13], medium (13, 24], large (24, 40], extra-large (> 40)}
	Shape: s	{cylinder, box, sphere, long-slender, irregular}

4.4.1 GPR and IR Bayesian Network Models

The BN models of a GPR sensor (Fig. 4.2.a) and of the Agema Thermovision 900 IR [1] are implemented in this dissertation to compute the expected information

value of a path, (4.7), or a robot configuration, (4.8), and to fuse the sensors' measurements after they are obtained. These BN models are learned from a database of sensor measurements produced by the demining simulation described in the previous section, using the approach presented in [1] and reviewed in Section 2.2. Because they rely on different operating principles and may function in different modes, these demining sensors are more or less effective depending on the environmental conditions. IR sensors, for example, detect anomalies in infrared radiation that is either emitted by mines, soil, or vegetation. Based on the location of the sensor, the radiation data can be processed to build an image of an horizontal area and to estimate the depth of the object therein for depths up to 12 cm. The mode m_{IR} influences the measured target features and is uniquely determined by its height above the ground. Therefore, airborne IR sensors typically obtain only cursory measurements of size z and shape s for shallow-buried objects. Because they rely on temperature variations, their performance also is highly influenced by illumination (time of day) i, weather w, vegetation g, and soil properties s_r, s_c, and s_u, with ranges described in Table 4.1.

GPR sensors emit radio waves that penetrate the ground and process their reflections at the boundaries of materials characterized by different refraction indexes. Images of underground vertical slices and of any objects buried within are obtained over the field of view, $\mathcal{S}$, by sensing discontinuities in electrical properties. The measured size z_m, shape s_m, and depth d_m of an underground object can be obtained from these images through signal processing techniques [85], such as edge extraction. The frequency of the radio wave and its bandwidth determine the search mode m_{GPR}. Since penetration depth increases at lower frequencies and image resolution improves at higher frequencies, the optimal GPR mode depends on target features F, and on the environmental conditions E shown in Table 4.1. For example, very high frequencies may be required in the presence of ground discontinuities to overcome

the so-called ground-bounce effect (GBE) [86]. By providing complementary information about the targets, the GPR measurements can significantly improve target classification through feature-level fusion with prior IR data [1,86]. Thus, after τ^* is planned and executed, the GPR and IR BN models are used to estimate F_i and y_i for each target intersected by $\mathcal{S}$ along τ^* from the *a-posteriori* evidence set $\mathcal{E}_i$ comprised of fused GPR and IR measurements, and of the operating and environmental conditions, m_{GPR_i}, m_{IR_i}, and E_i, as explained in Section 2.2 and [1].

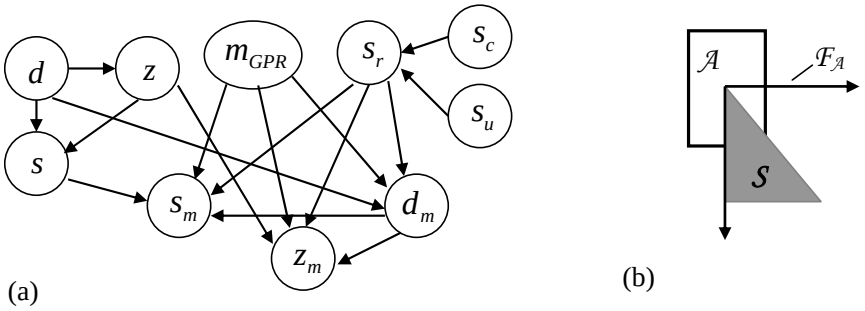


FIGURE 4.2: Example of BN model (a), adapted from [1], and geometric characteristics (b) of a GPR robotic sensor.

4.4.2 GPR Information Roadmap Method

Since the GPR sensor has limited field of view $\mathcal{S}$ and is installed on a ground robot, its measurements depend on the robot path in $\mathcal{W}$. The robot has a finite geometry $\mathcal{A}$ that must be taken into account in order to avoid natural or man-made obstacles (e.g., water bodies, trees, buildings) in $\mathcal{W}$. An example of robotic sensor geometry is shown in Fig. 4.2.b. As explained in Chapter 1, the position and orientation of every point in $\mathcal{S}$ and $\mathcal{A}$ can be specified using one configuration vector, $\mathbf{q}$, containing the coordinates and orientation of $\mathcal{F}_{\mathcal{A}}$ with respect to $\mathcal{F}_{\mathcal{W}}$. Airborne IR measurements are processed to obtain a map of the geometries and locations of potential obstacles B and targets T in $\mathcal{W}$. Together with the environmental information and the GPR BN model in Fig. 4.2.a, the IR measurements are also used to compute the sampling

PDF in (4.9), and to generate a set of N_m milestones by means of the hybrid sampling strategy in (4.10). N_m is defined by the user based on the size and complexity of $\mathcal{W}$, $\mathcal{CB}$, and $\mathcal{CT}$. The number of configurations $N_{\mathbf{q}}$ used to discretize $\mathcal{C}$ (algorithm in Appendix B) depends on the size of $\mathcal{W}$, and on the ranges of other degrees-of-freedom. In this chapter, the range $\theta \in [0,\ 7\pi/4]$ is discretized in eight intervals, and x and y are discretized in bins. For example, for the workspace in Fig. 4.6, $N_{\mathbf{q}} = 56.68{\cdot}10^3$ and $N_m = 900$.

The multivariate Gaussian λ_b can be generated using the Matlab® function *mvnrnd* [92], as follows: $\lambda_b = mvnrnd(\mu, \sigma)$, where the mean is $\mu = (x, y)$, and σ is a diagonal covariance matrix whose elements are chosen based on the width of the narrow passages from the interval $[0.5,\ 2]$. The user-defined parameters v_1 and v_2 in the sampling strategy, (4.10), determine the relative frequency of milestones sampled from narrow passages and from high information-valued targets, respectively. For example, the values used for minefields with a high density of obstacles and narrow passages (e.g., Fig. 4.8.a) are $v_1 = 0.1$ and $v_2 = 0.3$. Similarly, the weights w_V and w_D in the reward function, (4.1), represent a trade-off between the value of the measurements and the distance traveled, which typically is to be minimized to conserve time and energy. Thus, they depend on the application and on the units of V and D. Values of w_V and w_D in the intervals $[0.5,\ 1]$ and $[0,\ 0.5]$, respectively, were investigated in this chapter, ultimately selecting $w_V = 0.9$ and $w_D = 0.1$ for the demining application.

The weighting matrix W in (4.11) is chosen to scale the Eucledian distance based on the relative importance of the translational versus the rotational distance which, in turn, depends on the complexity of $\mathcal{CB}$ [39]. In this chapter, W is a diagonal matrix with elements $(0.9,\ 0.9,\ 0.1)$. Together with computational requirements, $\mathcal{CB}$ also determines the number of candidate neighbor nodes k considered by the local planner, which in this chapter takes integer values in the interval $[5,\ 20]$. In

most of the simulations shown in Section 4.5, $k = 6$. However, for high-density obstacles minefields (Fig. 4.8.a) the best value is found to be $k = 15$. For these parameters and the large workspace in Fig. 4.6, the maximum running time of the IRM algorithms was $1.5{\cdot}10^4$ sec for the learning phase, and 155 sec for the query phase, on an Intel T2060 1.6 GHz CPU computer, with 1.00 GB of RAM.

4.5 Results

The information roadmap method of a robotic GPR is tested on a variety of minefields exhibiting low-to-high densities of targets, obstacles, and narrow passages, as well as non-uniform soils, weather, and other environmental conditions. In Section 4.5.1, simple examples are used to illustrate and motivate the IRM approach, demonstrating the influence of the target geometries and prior information on the sensor path. In Section 4.5.2, the performance of IRM is compared to that of classical shortest-path solutions, complete-coverage paths, and randomized searches. Let $\Delta N_y(\tau)$ denote the number of targets that are correctly classified after fusing GPR and IR measurements along τ, minus the number of targets that were correctly classified based solely on IR measurements. Then the *classification efficiency* of a path τ is defined as the number of correctly-classified targets per distance traveled, $\eta_y(\tau) \equiv \Delta N_y(\tau)/D(\tau)$. After the path is executed and $M(\tau)$ becomes available, the *actual* entropy reduction, denoted by $\hat{V}(\tau)$, can be computed, and the metric $\hat{V}(\tau)/D(\tau)$ can be used to assess the actual information value per distance traveled along a path τ. The results in Section 4.5.2 indicate that IRM outperforms existing approaches by up to one order of magnitude. Also, as shown in Section 4.5.2, IRM can be applied to plan the path of non-overpass capable GPR platform, which must avoid collisions with mines as well as obstacles in order to prevent loss of the robotic sensor.

4.5.1 Influence of Prior information and Workspace Geometry on the Sensor Path

In this section, a series of simple examples are used to demonstrate why the geometric characteristics of the problem, together with prior information, must be taken into account in planning the sensor path. In every example, the IRM sensor path τ^* is illustrated by plotting sample sensor configurations on the workspace. Other hypothetical paths are schematized by dashed lines for comparison. The first example in Fig. 4.3 illustrates that both the location and geometry of targets and obstacles in $\mathcal{W}$ must be accounted for in planning the path of a robotic sensor. Suppose the sensor must travel from $\mathbf{q}_0$ to $\mathbf{q}_f$, and $\mathcal{W}$ contains two obstacles (black) and four equally-important targets (diagonal pattern) (Fig. 4.3). Although two obstacle-free paths τ^1 and τ^* of approximately the same distance can be found from $\mathbf{q}_0$ to $\mathbf{q}_f$, the path τ^1 (dashed line in Fig. 4.3) allows the GPR sensor to only make measurements of one target. Whereas, the IRM path τ^* allows the GPR sensor to make measurements of three of the targets in $\mathcal{W}$.

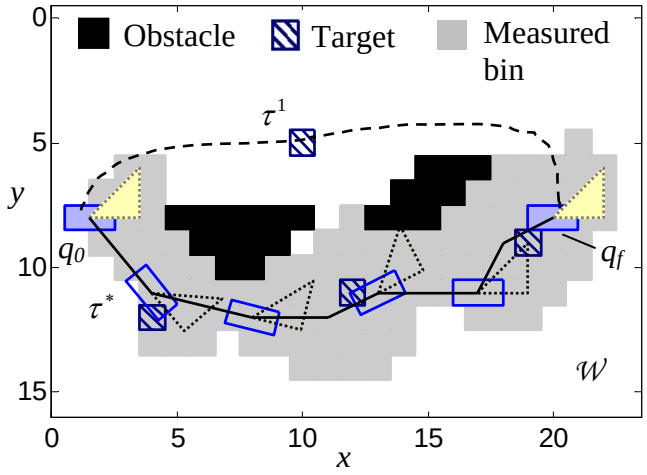
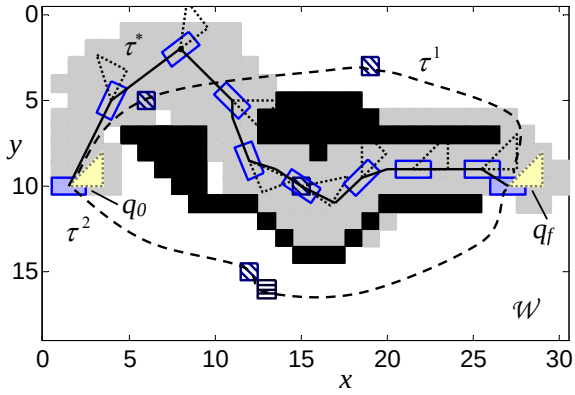


FIGURE 4.3: Influence of target presence on τ^*.

Besides accounting for targets' geometries and locations, IRM also accounts for the expected information value of the measurements that can be obtained from them,

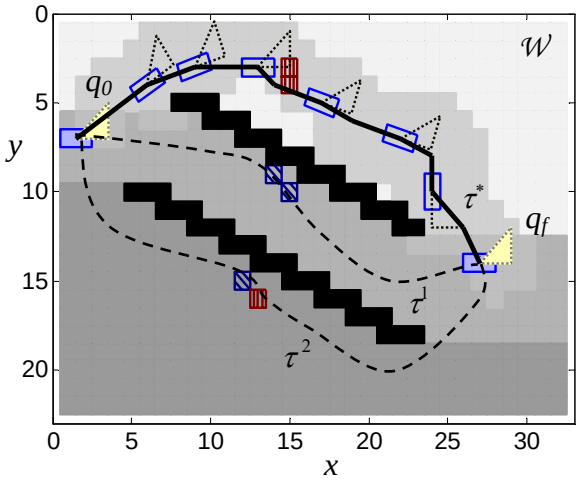
(4.4). Consider another simple example (Fig. 4.4) in which there exist three paths, τ^1, τ^2, and τ^*, that all allow the sensor to visit two targets by traveling approximately the same distance. Based on prior IR measurements, however, the information value of individual targets is either medium (diagonal pattern) or low (horizontal lines), as shown in Fig. 4.4. The information value is discretized only for illustration purposes. If only the targets' locations and geometries were taken into account, these three paths would be considered equivalent. Instead, by maximizing the information profit, τ^* obtains a much higher classification efficiency than τ^1 or τ^2.



Path:	τ^*	τ^1	τ^2
$D(\tau)$	30.711	33.675	27.797
$V(\tau)$	0.6450	0.4850	0.2335
$\eta_y(\tau)$	0.0651	0.0297	0

Target with $0.1 < V < 0.2$
Target with $V \leq 0.1$
Obstacle
Measured bin

FIGURE 4.4: Influence of prior sensor measurements on τ^*.



Path	τ^*	τ^1	τ^2
$D(\tau)$	30.075	25.914	33.986
$V(\tau)$	1.0646	0.5053	0.5913
$\eta_y(\tau)$	0.0665	0.0386	0

Moisture: Dry Wet Saturated
Target with $0.1 < V < 0.2$
Target with $V \geq 0.2$
Obstacle
Measured bin

FIGURE 4.5: Influence of environmental conditions on τ^*.

The influence of the environmental conditions surrounding a target, E_i, on the sensor measurements is also accounted for by the value of information, (4.4). For example, the same targets buried in different soils lead to very different reduction of the uncertainty (or entropy) in y_i, depending on how favorable the conditions are to the IR and GPR sensors. In the example in Fig. 4.5, two types of targets are buried in three different soil, for a total of six targets. In this workspace, there are three paths from $\mathbf{q}_0$ to $\mathbf{q}_f$ of approximately equal distance, that each visit the same two targets buried in soils with different moisture. One target has high information value in saturated soil, and two targets have high information value in dry soil. As shown in Fig. 4.5, by visiting targets with high information value, the GPR sensor obtains much higher classification efficiency along τ^* (Fig. 4.5). Therefore, the environmental conditions influence the sensor path by making targets more or less valuable to the GPR, and IRM finds the path that enables the most valuable GPR measurements.

4.5.2 IRM Efficiency Comparison and Results

This section summarizes the results obtained by testing the IRM approach on a variety of minefields with various sizes, geometries, and environmental conditions. A representative example of GPR path computed by IRM is shown in Fig. 4.6 for a 64×108 (bin) minefield with 755 buried objects that include mines and clutter, polygonal obstacles, several narrow passages, and heterogeneous environmental conditions (not shown for simplicity). The bins measured by the GPR along τ^* (solid black line) are show in grey, and the resulting path efficiency is $\eta_y = 0.0913$. These results show that the robotic GPR is capable of navigating in an obstacle-populated workspace, and through narrow passages, in order to make measurements from targets with high information value (plotted in red and magenta in Fig. 4.6) with minimium distance.

The average efficiency of IRM paths is compared to that of existing sensor path-

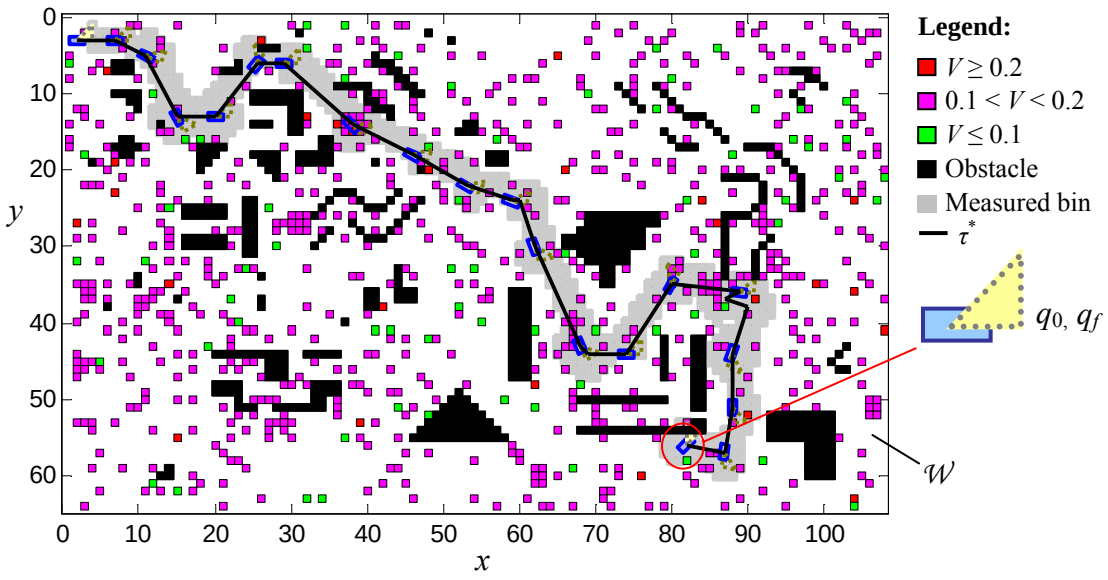


FIGURE 4.6: Robotic GPR sensor path obtained by information roadmap method (IRM).

planning methods, namely, shortest-path, complete-coverage, and randomized-search strategies, that are implemented here for the same robotic GPR sensor. The shortest-path search is a classical robot path-planning strategy that is applied in the query phase (Section 4.3) in order to determine the value of the efficiency metric η_y when the presence of targets is not taken into account. By this approach, the roadmap G, obtained in Section 4.2, is searched for the path of minimum distance, τ_{short}, with an A^*-type algorithm that uses the straight-line distance as heuristic function, and does not consider the expected information value. Then, the GPR is turned on in a fixed mode along the path, and known environmental conditions are used as evidence in the GPR BN model to infer the targets' classification from fused GPR-IR measurements.

Coverage path-planning algorithms play an important role in robotic sensing, because they emphasize the space swept by the robot's sensor [37]. As pointed out in [37], one of the most significant sensor path planning results is a planner that

generates a path that completely covers the obstacle-free space (e.g., see [2,3,93,94]). For comparison, in this chapter a complete-coverage path τ_{cover} is obtained for the robotic GPR in Fig. 4.2 by computing a lawn-mower-type path. This path consists of back-and-forth motions that avoid collisions between $\mathcal{A}$ and the obstacles B, while making measurements from the entire free-configuration space $\mathcal{C}_{free}$ with $\mathcal{S}$. Since computing the shortest complete-coverage path is known to be NP-hard [37], an *ad-hoc* solution to τ_{cover} is sought by placing milestones inside narrow passages, and near the boundaries of the C-obstacles and of the configuration space $\mathcal{C}$, denoted by $\partial\mathcal{CB}_i$ and $\partial\mathcal{C}$, respectively. The milestones inside narrow passages are obtained by the bridge test [38], and those near $\partial\mathcal{CB}_i$, $i \in I_T$, are obtained by a Gaussian sampler [40]. The milestones near $\partial\mathcal{C}$ are placed regularly spaced at a distance d_l that is based on the projection of $\mathcal{S}$ onto $\partial\mathcal{C}$. The set of milestones is ordered by increasing x and y coordinates, as to reproduce a lawn-mower path in an obstacle-free workspace. Then, the local planner described in Section 4.2 is used to connect the ordered milestones by collision-free straight-line segments. When the local planner fails (say in the x-direction), a milestone is inserted in the ordered list by increasing the other coordinate (say y) by a distance d_l. The new milestone is then connected to the nearest milestone in the list in the $-x$ direction, and so on. The result is a complete-coverage path, such as the one shown in Fig. 4.7.b, which covers $\mathcal{C}_{free}$, as illustrated by the grey bins measured by the GPR. The average efficiency of complete-coverage paths is shown in Table 4.2.

A popular approach in robotic demining and UXO clearance is to drive the robot in the minefield using a randomized search [3]. By this approach, a robot moves along a simple path, such as a straight line, until an obstacle is met, and then rotates a random amount before continuing along another simple path, while the on-board sensor is on at all times to detect targets along the path [3]. Using the roadmap G developed in Section 4.2, a random-search path, τ_{rand}, can be obtained by randomly

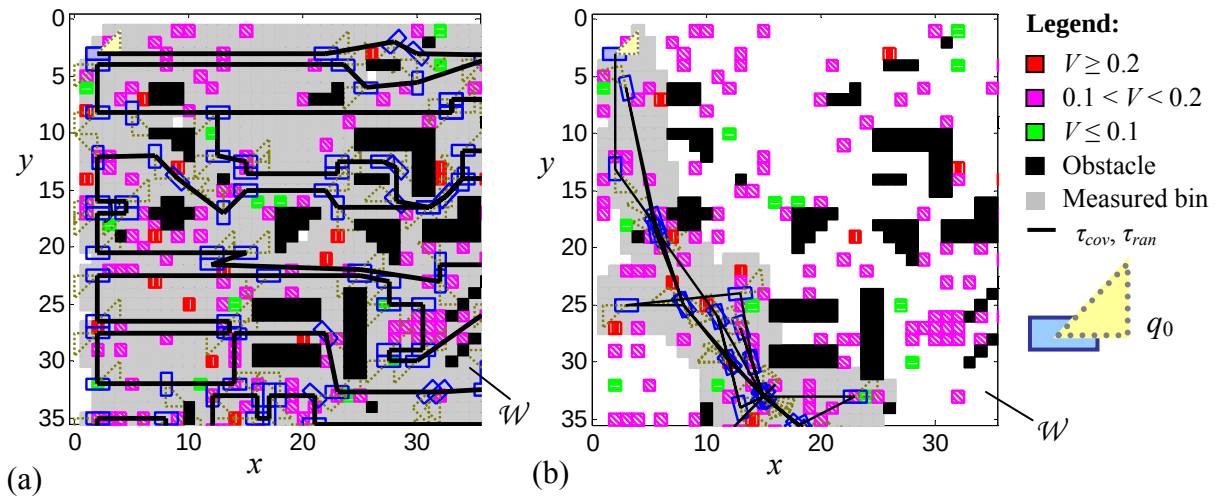


FIGURE 4.7: Robotic GPR sensor path obtained by complete coverage (a), and random search (b).

selecting a node from the list of neighbors of c_0 in G, and then repeating the process for every new node, until a pre-defined number of adjacent milestones connected by collision-free paths (arcs in G) is obtained, and the path executed. Although this approach does not optimize distance or information value in the query phase, it uses the information roadmap developed in Section 4.2. Therefore, the resulting path τ_{rand} guarantees obstacle avoidance by $\mathcal{A}$, and $\mathcal{S}$ is more likely to measure important targets than if deployed by a completely random strategy [3]. The four deployment strategies are applied to various minefields, for different values of $\mathbf{q}_0$ and $\mathbf{q}_f$, to obtain a representative average path efficiency η_y. The results are summarized in Table 4.2, and illustrate that IRM clearly outperforms the other methods leading, for example, to an average path efficiency three times greater than that of complete coverage, and one order-of-magnitude greater than that of shortest-path deployment. Additionally, by utilizing the information roadmap developed in Section 4.2, the other methods, in particular complete coverage, can be potentially very useful when measurements are required from the entire minefield or when the final configuration, $\mathbf{q}_0$, is not specified.

Since the sensor path efficiency depends on the characteristics of the workspace,

Table 4.2: Average sensor path efficiency.

Metric:	Deployment Method			
	IRM	Shortest Path	Random Search	Complete Coverage
$D(\tau)$	77	58	280	1491
$\hat{V}(\tau)/D(\tau)$	0.119	0.0463	0.0243	0.0432
$\eta_y(\tau)$	0.2093	0.0175	0.0504	0.0693

extensive numerical simulations were performed to obtain average efficiency metrics for different environmental conditions, and densities of obstacles and targets. Figure 4.8 illustrates two representative examples of workspaces that are considered to have high (a) or low (b) obstacle density. The target density is held constant in this study. As shown in Fig. 4.9, the path efficiency of IRM is significantly higher than that of shortest-path, random search, or complete coverage deployments, for both levels of obstacle density. The influence of environmental conditions on the method's efficiency is investigated by considering a minefield with the same geometric characteristics as the high obstacle-density example (Fig. 4.8.a), but with two typologies of environmental conditions, shown in Fig. 4.10. The soil composition, moisture, and vegetation are plotted over the workspace in Fig. 4.10.a for mild conditions, and in Fig. 4.10.b for harsh conditions. As shown in Fig. 4.11, IRM achieves the best performance under both types of conditions, but its improvement compared to other methods is smaller for harsh environments, because the accuracy of the GPR measurements decreases regardless of the path. In fact, the shortest-path performance improves under harsh environmental conditions because, when all targets have low information value, the efficiency metric can still be optimized by minimizing distance.

In another study, the average path efficiency is evaluated for low, medium, and high target densities (Fig. 4.12), using a constant obstacle density. As shown in Fig. 4.13, the IRM efficiency is significantly higher than that of complete coverage and random search methods, for all levels of target density. The most significant

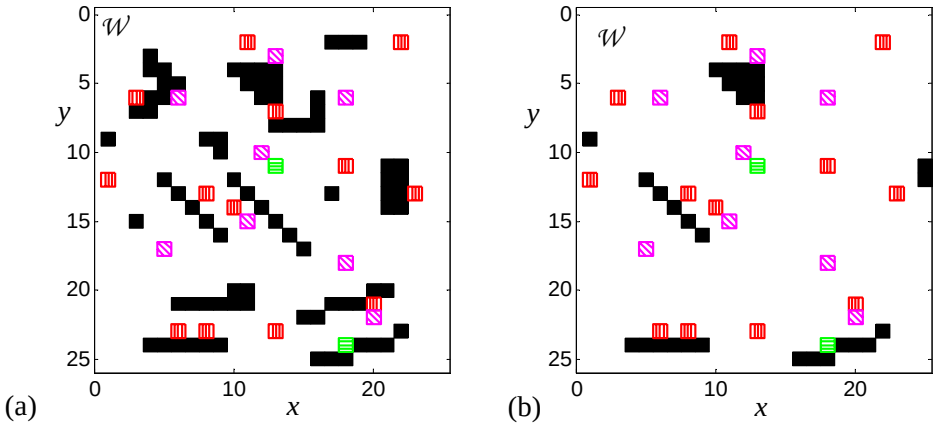


FIGURE 4.8: Examples of workspace with high (a) and low (b) obstacle density.

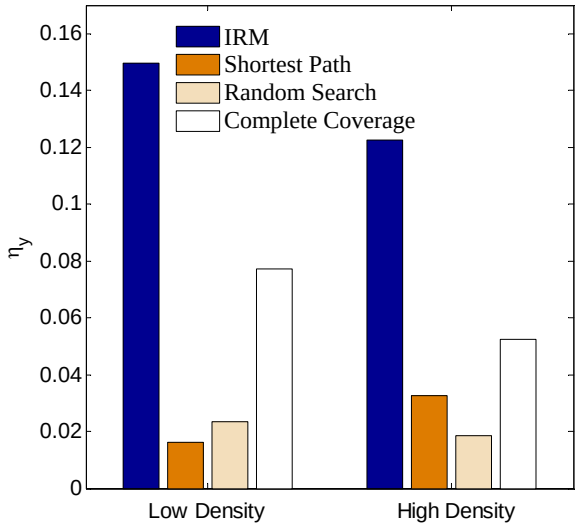


FIGURE 4.9: Influence of obstacle density on GPR sensor path efficiency.

improvement is obtained in minefields with high target density, because the sensor is able to visit more targets, taking full advantage of the proposed method. On the other hand, when the targets' density is low, their influence on the sensor path is considerably reduced, and therefore the efficiency of the IRM path approaches that of the shortest path. Conversely, the higher the target density, the closer the complete coverage efficiency will be to the IRM path, because the robotic sensor deployed by IRM will attempt to visit more targets within its reach.

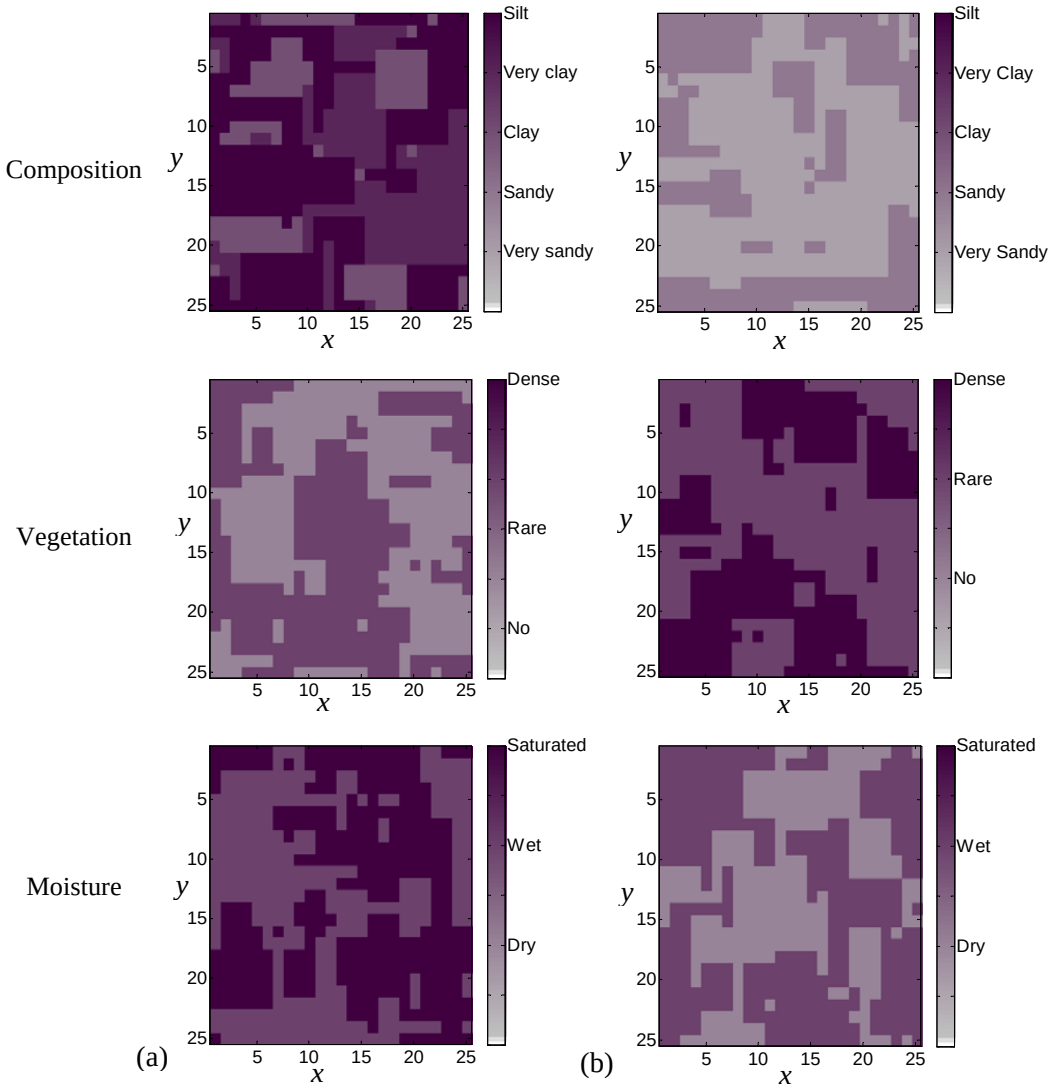


FIGURE 4.10: Example of mild (a) and harsh (b) environmental conditions for $\mathcal{W}$.

Based on the results in Figs. 4.9-4.13, it can be concluded that the efficiency of IRM is significantly higher than that of existing robotic sensor planning approaches applicable to geometric sensing, for a wide range of workspace conditions and characteristics. Another important application of the proposed method is planning the path of non-overpass capable robotic sensors that can be seriously damaged or even destroyed when driving over landmines [8]. For this type of robotic platforms, the set of targets T in $\mathcal{W}$ must be treated as an additional set of obstacles,

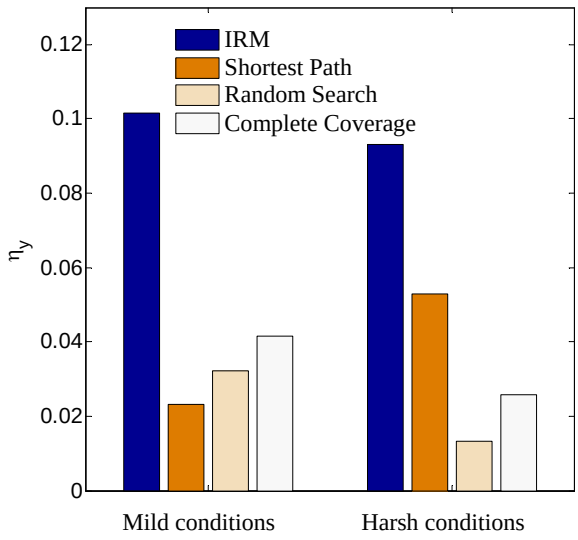


FIGURE 4.11: Influence of environmental conditions on GPR sensor path efficiency.

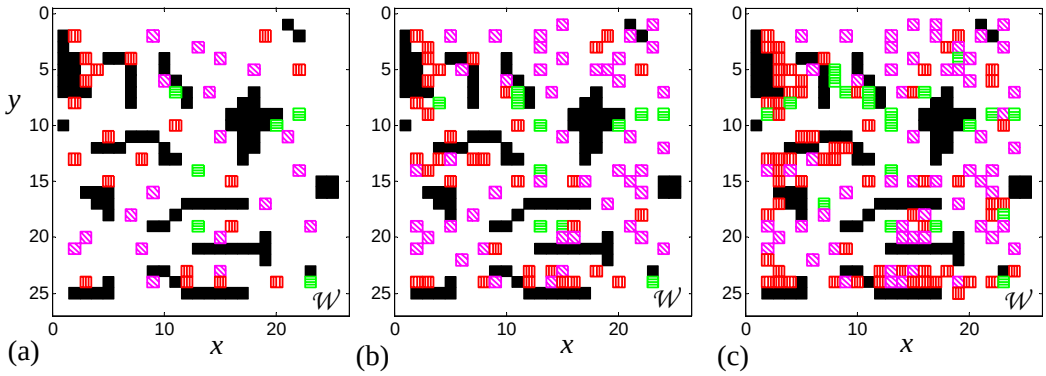


FIGURE 4.12: Example of low (a), medium (b), and high (c) target density.

$B := \{\mathcal{B}_1, \ldots, \mathcal{B}_n\} \cup T$, to be avoided by $\mathcal{A}$ in case they are landmines. As shown by the examples in Fig. 4.14, the robotic GPR can still obtain measurements from the targets by navigating near them, such that they may be intersected by $\mathcal{S}$ but not by $\mathcal{A}$. By implementing the information roadmap developed in Section 4.2, both IRM and complete coverage methods allow a non-overpass capable GPR sensor to navigate the workspace and make measurements from targets in $\mathcal{W}$, including those inside narrow passages which further restrict the free configuration space (Fig. 4.14). As can be expected, the average efficiencies (Fig. 4.15) are lower than those obtained

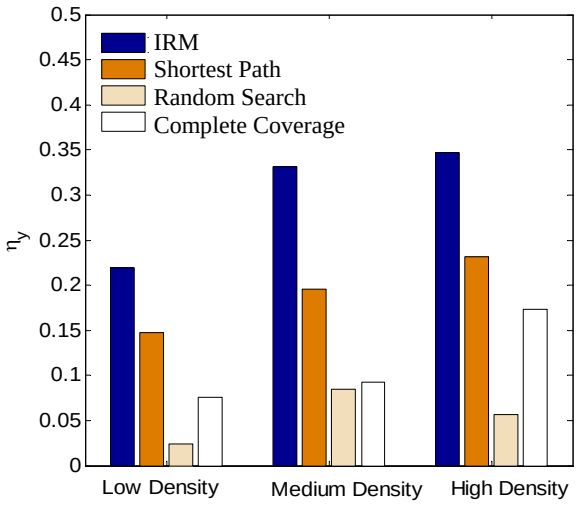


FIGURE 4.13: Influence of target density on GPR sensor path efficiency.

by overpass-capable platforms (Table 4.2), because the sensor may need to travel a longer distance in order to avoid platform collisions with the targets. However, compared to shortest path, random search, and complete coverage, IRM still improves the number of targets that are properly classified per unit distance (η_y) by up to one order of magnitude (Fig. 4.15).

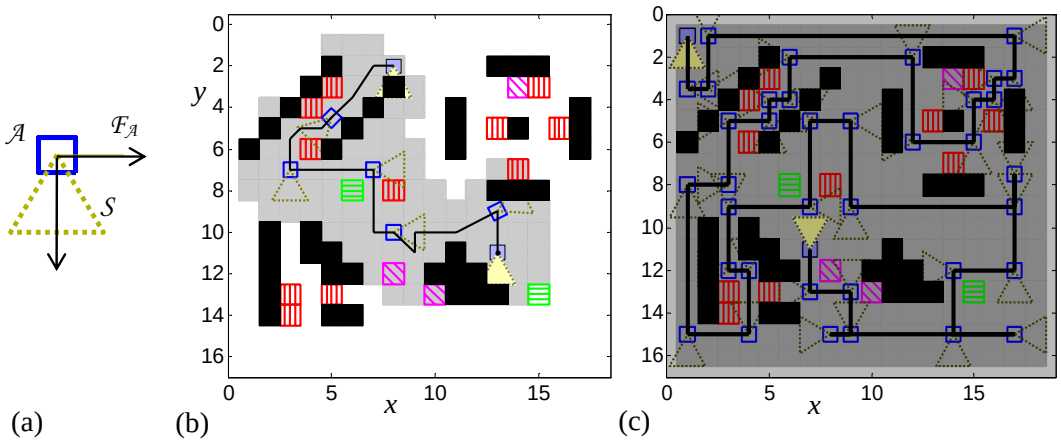


FIGURE 4.14: Examples of IRM (b) and complete-coverage (c) paths for a non-overpass capable robotic platform $\mathcal{A}$, equipped with an on-board GPR sensor with field of view $\mathcal{S}$ (a).

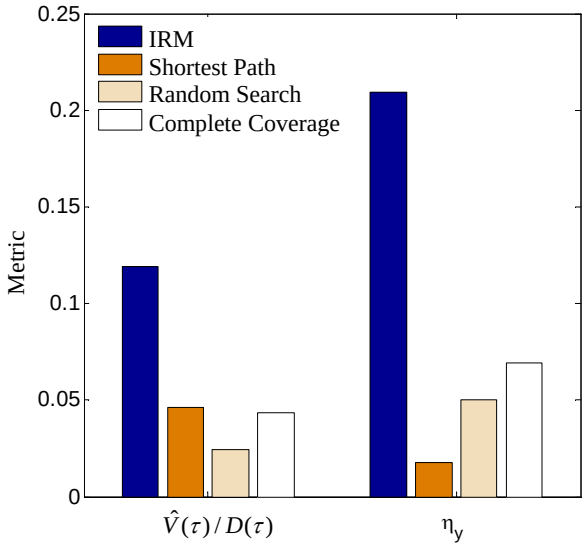


FIGURE 4.15: Average sensor path efficiency for the non-overpass capable GPR sensor in Fig. 4.14.a.

5

Online Information Potential Method for Robotic Sensor Networks

In the previous chapter, the information roadmap is utilized to generate the path for a single robotic sensor in a two-dimensional workspace. In this chapter, we extend this problem in the following aspects. Firstly, multiple robotic sensors are employed to detect and classify targets in a three-dimensional workspace populated by multiple obstacles. Secondly, prior information on targets and obstacles may be either available or unavailable *a priori*, and can become available online during the process. Finally the robotic sensor dynamics are taken into account, and the feedback control designed is integrated with the sensor path planning problem.

5.1 Robotic Sensor Model and Workspace

In this dissertation, we assume that the kinematics of the robotic sensor platform can be approximated by the unicycle model,

$$\begin{aligned} \dot{x} &= v\ cos(\theta) \\ \dot{y} &= v\ sin(\theta) \\ \dot{\theta} &= w \\ \dot{v} &= a \end{aligned} \tag{5.1}$$

where $[x\ y]^T$ is the position of the robot centroid, θ is its orientation, v is the linear velocity, w is the angular velocity, a is the linear acceleration, and $u = [a\ w]^T \in \mathcal{U}$ is the control vector. $\mathcal{U}$ represents the control space. The sensor can either be mounted at the center of the platform or any other point with arbitrary orientation. As an underactuated and nonholonomic system, it is difficult to convert (5.1) into an equation with a three-dimensional vector of double integrators [95]. However, with the analysis in [96], it is possible to define the control and convert it into two-dimensional space. The control law of the robotic sensor model is described in Section 5.5.

The workspace $\mathcal{W}$ is assumed to be a three-dimensional compact, bounded subset of a Euclidian space populated with obstacles and targets that are assumed to be convex right prisms where the base faces are parallel to the x-y plane. The vertical position of a target $\mathcal{T}_i$, is defined by a vector $[h_1\ h_2]^T$ representing the heights of its base faces and is illustrated in Fig. 5.1. The base shape of $\mathcal{T}_i$ and the values of h_1 and h_2 will decide the $\mathcal{CT}_i$ for a specific robotic sensor FOV. The obstacles can be defined in the similar way.

The robotic sensor platform $\mathcal{A}$ is also assumed to be a convex right prism with the same property as the target, and $h_1 \equiv 0$ since the robotic sensor always contacts the

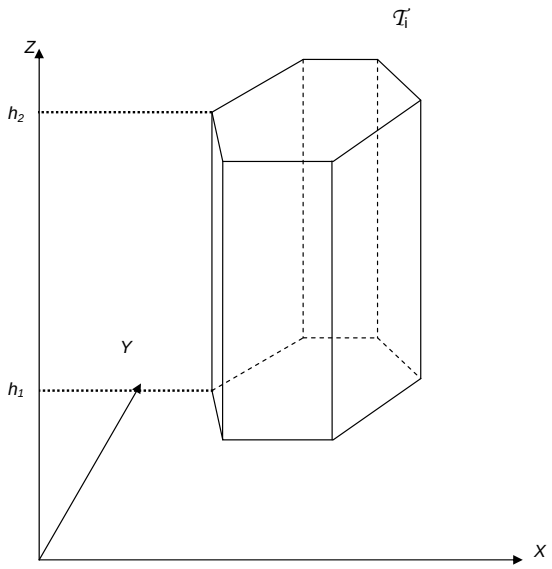


FIGURE 5.1: An example of target geometry and height.

ground. In this chapter, the geometry of the sensor FOV $\mathcal{S}$ is represented by a three-dimensional cone as shown in Fig. 5.2(a). The sensor FOV could be represented by a two-dimensional polygon that varies based on the target height. An example of the sensor FOV for a target is shown in Fig. 5.2(b). In this figure, the sensor FOV apex position in $\mathcal{F}_{\mathcal{A}}$ is $(0, 0, 5)$, and the target is assumed to be float in the air with minimal height $h_1 = 6.5$. The floating targets may have various applications, such as monitoring urban environment and underwater objects. The technical challenges lie in that the region in which the sensor can measure the target is not only decided by the robotic sensor configuration but also the height of the target.

An example of a bird view of the workspace is shown in Fig. 5.3. Let $\mathcal{A}_i$ and $\mathcal{S}_i$ represent the platform and FOV of the ith robotic sensor. $A = \{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_r\}$ and $S = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_r\}$ indicate the set of robotic sensor platforms and sensor FOVs in the workspace respectively. $\mathcal{F}_{\mathcal{A}i}$ is a moving Cartesian frame embedded in $\mathcal{A}_i$. Then, every point of $\mathcal{S}_i$ has a fixed position with respect to $\mathcal{F}_{\mathcal{A}i}$, and the configuration $\mathbf{q}_i = [x_i \ y_i \ \theta_i]^T \in SE(2)$ is used to specify the position $[x_i \ y_i]^T$ and orientation θ_i of

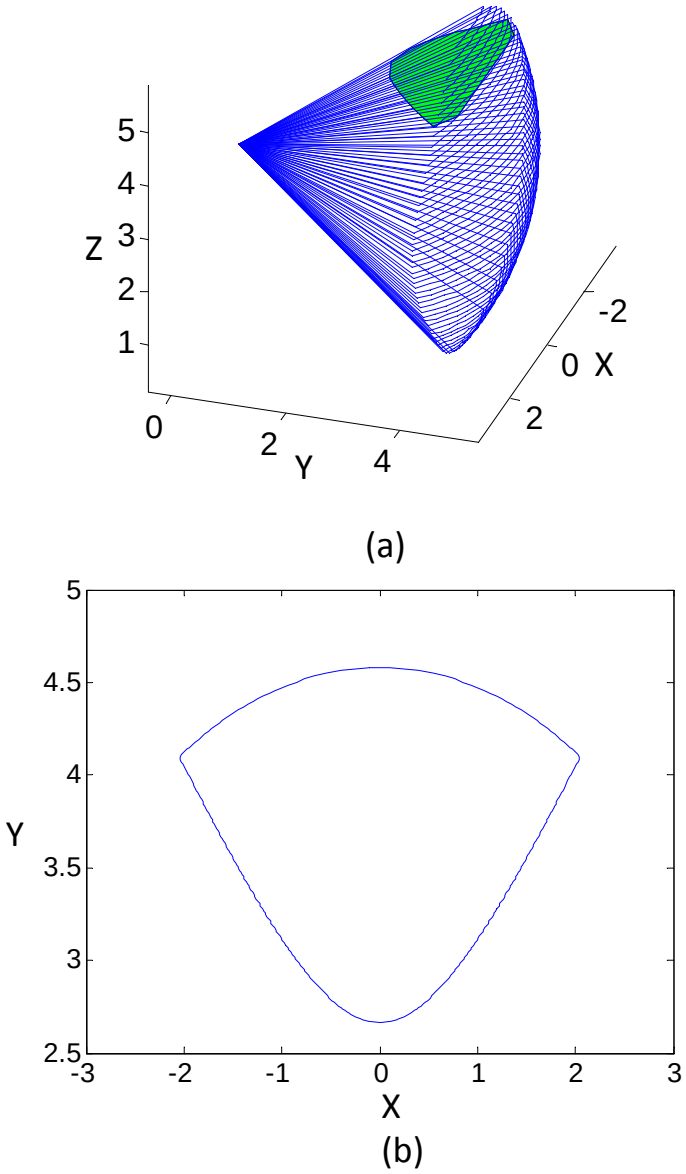


FIGURE 5.2: The FOV of a sensor with taper shape: (a) sensor FOV in three-dimensional space; (b) reduced sensor FOV in two-dimensional space.

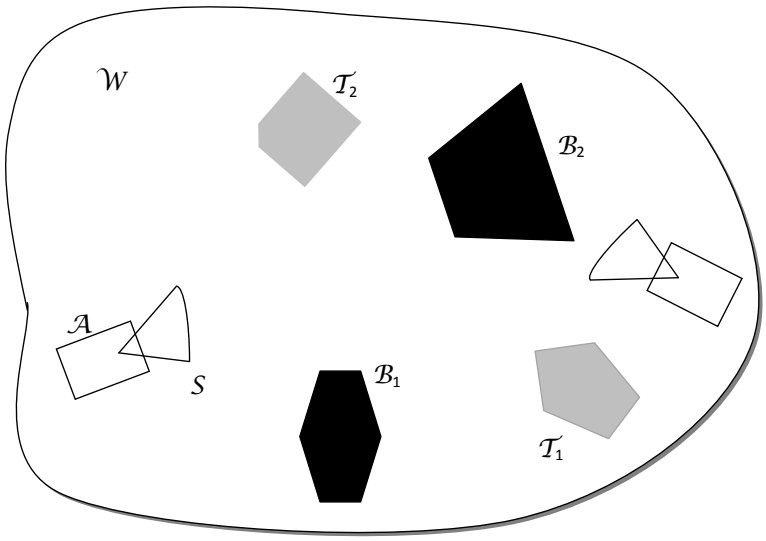


FIGURE 5.3: Relevant problem geometries and notation.

all $\mathcal{A}_i$ and $\mathcal{S}_i$ with respect to a fixed inertial frame $\mathcal{F}_\mathcal{W}$, embedded in $\mathcal{W}$. Obstacles and targets are also assumed to be fixed and rigid in $\mathcal{W}$, such that every point of $\mathcal{B}_i$, for $\forall i \in I_B$, and every point of $\mathcal{T}_j$, $\forall j \in I_T$, have a fixed position with respect to $\mathcal{F}_\mathcal{W}$. Let $\mathcal{C}$ denote the space of all possible robot configurations. Then, the path of the ith robotic platform's centroid is defined as a continuous map $\tau_i : [0, 1] \to \mathcal{C}$, with $\mathbf{q}_{i_0} = \tau_i(0)$ and $\mathbf{q}_{i_f} = \tau_i(1)$. $\mathbf{q}_{i_0}$ is the initial configuration, and $\mathbf{q}_{i_f}$ is the final configuration which is decided by the stopping time t_f. Since $\mathcal{S}_i$ is mounted on $\mathcal{A}_i$, the path τ_i determines the targets in $\mathcal{W}$ that can be measured by this robotic sensor, while traveling from $\mathbf{q}_{i_0}$ to $\mathbf{q}_{i_f}$. Then, the set of paths $\Gamma = \{\tau_1, \ldots, \tau_r\}$ determines the targets in $\mathcal{W}$ that can be measured and classified by the robotic sensor network starting at $\{\mathbf{q}_{1_0}, \ldots, \mathbf{q}_{r_0}\}$.

The robotic sensor network is deployed to measure and classify multiple fixed targets in $\mathcal{W}$, based on new (posterior) and prior sensor measurements, and environmental information. This information can either be obtained by global sensors such as an airborne sensor, or by a sensor mounted on a mobile platform that has been

previous deployed. Therefore, to optimize performance, the path planning algorithm must take into account the prior information, for correctly classifying targets in the workspace. Once a target $i \in I_T$ is detected, we assume that its location and geometry $\mathcal{T}_i$ become known, but its state $\mathbf{X}_i$ remains uncertain due to the random nature of the measurement process. Therefore, the information value of the ith target can be computed by one of the information functions in Chapter 3. Let V_i denote the expected utility of making additional measurements from $\mathcal{T}_i$, and it can be represented by the expected reduction in uncertainty associated with $\mathbf{Z}_i$, conditioned on the sensor mode and prior information.

Let the measurement set obtained by the r robotic sensors along their respective paths $\Gamma = \{\tau_1, \ldots, \tau_r\}$ be defined as $Z = \{\mathbf{Z}_j \mid \mathcal{T}_j \cap \mathcal{S}_i(\mathbf{q}_i) \neq \emptyset,\ \tau_i(s) = \mathbf{q}_i,\ s \in [0,1],\ \forall j \in I_T,\ \forall i \in I_A\}$, where $\mathcal{S}_i(\mathbf{q}_i)$ is the ith sensor FOV at configuration $\mathbf{q}$ in $\mathcal{W}$, and I_A is the index set for the sensor platforms. The detected obstacles and targets are referred to obstacles and targets that are detected by a global sensor or a robotic sensor up to the present time. For each detected obstacle and target, its geometry and location is assumed to be known. Then, the robotic sensors paths should be computed to achieve the following objectives: (i) explore the workspace to detect new targets and obstacles, (ii) assign targets to sensors based on the latest information on the workspace, (iii) maximize the information value of Z, (iv) avoid all obstacles and (v) robots in $\mathcal{W}$, while traveling the minimum total distance starting at $\{\mathbf{q}_{1_0}, \ldots, \mathbf{q}_{r_0}\}$. Thus, while every platform in the set A must avoid intersections (collisions) with other platforms and all obstacles in B, the sensor FOVs in the set S must intersect the geometries of the targets with high value of information in order to obtain additional measurements and improve the classification of the targets.

5.2 Information Potential Method

In this section, the potential field is generated by first assuming that only one robotic sensor is deployed in the workspace. When multiple robotic sensors are deployed in the workspace, the targets are assigned to these robotic sensors and the path of each robotic sensor is generated.

Let $U(\mathbf{q})_{rep}$ represent the repulsive potential to avoid collision between the robotic sensor and obstacles, and $U(\mathbf{q})_{att}$ represent the attractive potential to attract the robotic sensor to measure the targets. The sensor's potential function is sum of $U(\mathbf{q})_{rep}$ and $U(\mathbf{q})_{att}$ and is defined in (5.2),

$$U(\mathbf{q}) = U(\mathbf{q})_{rep} + U(\mathbf{q})_{att} \tag{5.2}$$

for $\mathbf{q} \in \mathcal{C}$. Since the target geometries are considered, $\rho_i^t(\mathbf{q})$, which is the distance between the ith target and $\mathbf{q}$, needs to be first computed to generate $U_i(\mathbf{q})_{att}$. The concept of C-targets, defined in Section 4.2, is utilized to describe the set of configurations in $\mathcal{C}_{free}$ for which the robotic sensor can take measurement on at least one target. $\rho_i^t(\mathbf{q})$ is computed as,

$$\rho_i^t(\mathbf{q}) = \min_{\mathbf{q}_i \in \mathcal{CT}_i} ||W \cdot (\mathbf{q}_i - \mathbf{q})|| \tag{5.3}$$

where $||\cdot||$ represents the Euclidian norm, and W is a diagonal and positive definite matrix representing the relative importance of changes in position and orientation. W is chosen by the user based on the trade off on the cost of translation and rotation. The information value of each target can be measured by various information metrics based on prior information. In this chapter we utilize mutual information as the information metric. As described in Chapter 3 the uncertainty of the hidden variable for the ith target, denoted by $\mathbf{X_i}$, given the prior information $\mathbf{e}$, can be represented

by the conditional entropy defined as,

$$H(\mathbf{X_i}|\mathbf{e}) = \sum_{\mathbf{x_i}\in\mathcal{X}} P(\mathbf{x_i}|\mathbf{e}) \log_2 P(\mathbf{x_i}|\mathbf{e}) \tag{5.4}$$

Hence, the expected decrease of uncertainty in $\mathbf{X}_i$ by a posterior measurement $\mathbf{Z_i^k}$ can be measured by the mutual information

$$\begin{aligned} I(\mathbf{X_i}; \mathbf{Z_i^k}|\mathbf{e}) &= H(\mathbf{X_i}|\mathbf{e}) - \mathbb{E}H(\mathbf{X_i}|\mathbf{Z_i^k}, \mathbf{e}) \\ &= H(\mathbf{X_i}|e) - \sum_{\mathbf{z_i^k}\in\mathcal{Z}} P(\mathbf{z_i^k}|\mathbf{e}) H(\mathbf{X_i}|\mathbf{z_i^k}, \mathbf{e}) \end{aligned} \tag{5.5}$$

In this chapter, the information value V_i of the ith target, estimated by $I(\mathbf{X_i}; \mathbf{Z_i^k}|\mathbf{e})$ in (5.5), is used to construct the potential function, such that the sensor path takes into account the expected utility of making measurement from $\mathcal{T}_i$ prior to visiting the target. The attractive potential function has the following properties.

1. $U_i(\mathbf{q})_{att}$ is an increasing function of the distance $\rho_i^t(\mathbf{q})$. When $\rho_i^t(\mathbf{q})$ goes to infinity, the potential converges to a finite value.

2. The potential generated by the target with the higher information value has the bigger distance of influence, where the distance of influence is defined by the inflexion point of the potential value as a function of $\rho_i^t(\mathbf{q})$.

To achieve these objectives, we propose an attractive function for the ith target that is defined as,

$$U_i(\mathbf{q})_{att} \triangleq \eta_2 \sigma V_i^a (1 - e^{-\frac{\rho_i^t(\mathbf{q})^2}{2\sigma V_i^a}}) \tag{5.6}$$

where η_2 is a scaling parameter representing the influence of targets, V_i is the information value of the ith target, and σ is the influence parameter which together with V_i and parameter a decides the influence distance of the ith target. It can be

shown that (5.6) satisfies the above properties. Let $\rho_i = \rho_i^t(\mathbf{q})$, $U_i(\rho_i)_{att}$ can also be regarded as a function of ρ_i. The first order derivative of $U_i(\rho_i)_{att}$ is,

$$g(\rho_i) = \frac{dU_i(\rho_i)_{att}}{d\rho_i} = \eta_2 \rho_i e^{-\frac{\rho_i^2}{2\sigma V_i^a}} \tag{5.7}$$

g is positive for all $\rho_i > 0$, which shows that $U_i(\rho)_{att}$ is an increasing function of ρ_i. Also from (5.6),

$$\lim_{\rho_i \to \infty} U_i(\rho_i)_{att} = \eta_2 V_i^a \tag{5.8}$$

so property 1 is satisfied. To demonstrate property 2, we compute the inflection point of $U_i(\rho_i)_{att}$. The second derivative of $U_i(\rho_i)_{att}$,

$$\frac{dg(\rho_i)}{d\rho_i} = \eta_2 (1 - \frac{\rho_i^2}{\sigma V_i^a}) e^{-\frac{\rho_i^2}{2\sigma V_i^a}} \tag{5.9}$$

is a monotonically decreasing function of ρ_i and setting it to zero will give us the influence distance of the ith target $\rho_i = \sqrt{\sigma V_i^a}$, which is an increasing function for V_i.

To guarantee that the value of the potential function in every C-target is a local minimum of U_{att}, we consider the total attractive potential at a configuration $\mathbf{q}$, defined as,

$$U(\mathbf{q})_{att} \triangleq \Pi_{i=1}^m U_i(\mathbf{q})_{att} \tag{5.10}$$

which is characterized by the gradient,

$$\nabla U_{att}(\mathbf{q}) = N_1 \mathbf{n}_1^T + \ldots + N_m \mathbf{n}_m^T \tag{5.11}$$

where

$$N_i = \Pi_{j=1, j \neq i}^m U_j(\mathbf{q})_{att} \eta_2 e^{-\frac{\rho_i^t(\mathbf{q})^2}{2\sigma V_i^a}} \rho_i^t(\mathbf{q}) \quad i = 1, \ldots, m \tag{5.12}$$

and $\mathbf{n}_i^T = \nabla \rho_i^t(\mathbf{q})$. $\mathbf{n}_i^T$ is a unit vector pointing from the closest configuration on $\mathcal{CT}_i$ toward $\mathbf{q}$.

The repulsive potential for the ith obstacle is defined as,

$$U_i(\mathbf{q})_{rep} \triangleq \begin{cases} \frac{1}{2}\eta_1(\frac{1}{\rho_i^b(\mathbf{q})} - \frac{1}{\rho_0})^2 U_{att}(\mathbf{q}) & \text{if} \quad \rho_i^b(\mathbf{q}) \leq \rho_0 \\ 0 & \text{if} \quad \rho_i^b(\mathbf{q}) > \rho_0 \end{cases} \tag{5.13}$$

where η_1 is a scaling parameter showing the influence of obstacles, ρ_0 is the influence distance of obstacles, and $\rho_i^b(\mathbf{q})$ is the distance between $\mathbf{q}$ and the ith obstacle. Then the total repulsive potential is defined as,

$$U(\mathbf{q})_{rep} \triangleq \sum_{i=1}^{n} U_i(\mathbf{q})_{rep} \tag{5.14}$$

In order to make sure that the C-target inside an obstacle's influence distance is measurable by the robotic sensor, the term $U_{att}(\mathbf{q})$ is added in (5.13).

Let I_{ρ_0} be the index set for the obstacles that have configuration $\mathbf{q}$ inside their influence distance. Then the gradient of $U_{rep}(\mathbf{q})$ is

$$\nabla U_{rep}(\mathbf{q}) = \sum_{i \in I_{\rho_0}} M_i^B \mathbf{n}_i^B + \sum_{j=1}^{m} M_j^T \mathbf{n}_j^T \tag{5.15}$$

where,

$$M_i^B = \eta_1 \left(\frac{1}{\rho_i^b(\mathbf{q})} - \frac{1}{\rho_0} \right) \frac{U(\mathbf{q})_{att}}{(\rho_i^b(\mathbf{q}))^2} \qquad i \in B_{in}$$

$$M_i^T = \left(\sum_{i \in I_{\rho_0}} \frac{1}{2}\eta_1 (\frac{1}{\rho_i^b(\mathbf{q})} - \frac{1}{\rho_0})^2 \right) \Pi_{i=1, i \neq j}^{m} U_i(\mathbf{q})_{att} \eta_2 e^{-\frac{\rho_i^t(\mathbf{q})^2}{2\sigma V_i^a}} \rho_i^t(\mathbf{q}) \qquad i = 1, \ldots, m \tag{5.16}$$

and $\mathbf{n}_i^B = \nabla \rho_i^b(\mathbf{q})$. $\mathbf{n}_i^B$ is a unit vector pointing from $\mathbf{q}$ toward the closest configuration on $\mathcal{CB}_i$. The gradient of the total potential function is

$$\begin{aligned} \nabla U(\mathbf{q}) &= \sum_{i=1}^{m} N_i \mathbf{n}_i^T + \sum_{i \in I_{\rho_0}} M_i^B \mathbf{n}_i^B + \sum_{j=1}^{m} M_j^T \mathbf{n}_i^T \\ &= \sum_{i \in I_{\rho_0}} M_i^B \mathbf{n}_i^B + \sum_{i=1}^{m} (N_i + M_i^T) \mathbf{n}_i^T \end{aligned} \tag{5.17}$$

and can be used to control the robotic sensor's movement in the configuration space as shown in Section 5.5.

5.3 Multiple-sensor Path Planning Problem

In the online sensor path problem, multiple robotic sensors are working together to measure a subset of targets deployed in a partially-known workspace. Unlike the robotic sensor in Chapter 4, we assume that each robotic sensor may be equipped two sensors, so that it can be used to detect unknown obstacles and targets. One sensor has a large FOV and obtains accurate obstacle information but cursory target measurement. Another sensor has a small FOV and obtain additional measurements from the targets to improve their classification.

The process of this sensor path planning problem is divided into a sequence of sub-processes, which begin when a new target or obstacle is detected or an additional measurement on a detected target becomes available. Each sub-process involves the following key steps. Firstly, each detected target is assigned to the robotic sensor with smallest distance. Then the robotic sensors are set into two different modes based on the results of target assignment. The robotic sensors with at least one target assigned to are set in classification mode to take additional measurements form these targets to improve their classification. The robotic sensors without any target assigned to are set in exploration mode to detect unknown obstacles and targets in the workspace. A switched feedback control is integrated in this path planning problem to control each robotic sensor based on the distance between the robotic sensor and its closest target.

It is assumed that all targets detected up to the present time are assigned to sensors based on distance as follows. Let ρ_{ij} represent the distance between the ith target and the jth robotic sensor. $N(j)$ be the set containing targets assigned to the

jth robotic sensor

$$N(j) = \{\mathcal{T}_i \mid \rho_{ij} < \rho_{ik}, i \in I_T, \forall k \neq j\} \tag{5.18}$$

When $N(j) = \emptyset$ the jth robotic sensor is in exploration mode. A potential field is constructed for the robotic sensor based on a virtual goal which is generated as follows. Let $\mathcal{C}_{det}$ be the region in $\mathcal{C}$ that has been detected by the robotic sensor network, then the undetected region is $\mathcal{C}_{udet} = \mathcal{C} \setminus \mathcal{C}_{det}$. Based on a uniform distribution on $\mathcal{C}_{udet}$, k congratulations are randomly sampled in $\mathcal{C}_{udet}$, denoted by $\{\mathbf{s}_1, \ldots, \mathbf{s}_k\}$, where k is a positive constant determined by the user. The distance between the virtual goal and the jth robotic sensor is computed by

$$\rho^v(\mathbf{q}) = \min_{i=1,\ldots,k} \|W \cdot (\mathbf{s}_i - \mathbf{q}_j)| \tag{5.19}$$

where $||\cdot||$ represents the Euclidian norm, and W is a diagonal and positive definite matrix representing the relative importance of changes in position and orientation. W is chosen by the user based on the trade off on the cost of translation and rotation. A new attractive potential U^v_{att} generated by this virtual goal is

$$U^j(\mathbf{q})_{att} = \frac{1}{2}\eta_4 \rho^v(\mathbf{q})^2 \tag{5.20}$$

where η_4 is scaling parameter. Then the total potential function for the jth robotic sensor in the exploration mode is

$$U(\mathbf{q}) = U(\mathbf{q})_{rep} + U^j(\mathbf{q})_{att} \tag{5.21}$$

where $U(\mathbf{q})_{rep}$ is the repulsive potential function defined in (5.14).

When $N(j) \neq \emptyset$, the jth robotic sensor is set in the classification mode. Detected targets in $N(j)$ is utilized to construct the attractive potential function, denoted by $U^j(\mathbf{q})_{att}$, for the jth robotic sensor, and the all the detected obstacles are used to construct the repulsive potential function by obstacles.

Since multiple robotic sensors are moving simultaneously, the robotic sensor also needs to avoid collision with other robotic sensors. A potential value U_r^j generated by all the other robotic sensors is added to the potential field of jth robotic sensor to avoid collision between robotic sensors.

$$U_r^j(\mathbf{q}) = \sum_{k \neq j} U_{rk}^j(\mathbf{q}) \tag{5.22}$$

where,

$$U_{rk}^j(\mathbf{q}) = \begin{cases} \frac{1}{2}\eta_3(\frac{1}{\rho_{jk}^r(\mathbf{q})} - \frac{1}{\rho_0})^2 & \text{if} \quad \rho_{jk}^r(\mathbf{q}) \leq \rho_0 \\ 0 & \text{if} \quad \rho_{jk}^r(\mathbf{q}) > \rho_0 \end{cases} \tag{5.23}$$

η_3 is a scaling parameter, and $\rho_{jk}^r(\mathbf{q})$ is the distance between the jth and kth robotic sensor. The total potential value for the jth robotic sensor is

$$U^j(\mathbf{q}) = U(\mathbf{q})_{rep} + U^j(\mathbf{q})_{att} + U_r^j(\mathbf{q}) \tag{5.24}$$

This proposed potential field is not free of local minima, therefore an algorithm needs to be provided to help the robotic sensor escape from the local minimum. Various methods have been proposed to overcome this limitation of potential field. When the workspace dimension is low, such as two or three dimensions, robots can escape a local minimum by filling the well. However, when the workspace dimension is high, this method is no longer feasible due to its computational complexity. Then other methods, usually random methods, are used to help the robots escape the local minima, for instance randomized path planning [50]. In this chapter, the potential function is utilized to generate a local information roadmap around the local minimum to move the robotic sensor away from the local minimum. Unlike normal randomized path planning, information roadmap in our method considers the probability of measuring targets after escaping the local minima during milestones sampling. This makes it more applicable in problems with multiple targets. Since PRM has

also been proved applicable to high-dimensional workspaces, IPM combines potential field and information roadmap to sensor navigation in high-dimensional workspaces.

Suppose the jth robotic sensor arrives at a local minimum, called $\mathbf{u}_{loc}$. The probability density function for sampling milestones at $\mathbf{q}$ is defined as,

$$f_j(\mathbf{q}) = \begin{cases} \frac{e^{-U^j(\mathbf{q})}}{\int_{\mathcal{E}} e^{-U^j(\mathbf{q})} d\mathbf{q}} & \mathbf{q} \in \mathcal{E} \\ 0 & \mathbf{q} \notin \mathcal{E} \end{cases} \tag{5.25}$$

where $\mathcal{E} \subset \mathcal{C}_{det}$ is a randomly generated subspace around $\mathbf{u}_{loc}$. By defining $f_j(\mathbf{q})$ as shown in (5.25), configurations in $\mathcal{E}$ which are close to targets with high information value and far away from obstacles have high probability to be sampled. Then a specific number of milestones $\{c_1, c_2, \ldots, c_k\}$ can be sampled by *Direct Methods* [97]. Call $\mathbf{u}_{loc}$ as c_0, the set $C = \{c_0, c_1, c_2, \ldots, c_k\}$ together with a local planner is used to construct the roadmap. While the normal roadmap construction method first constructs the roadmap with sampled milestones and then tries to connect the initial configuration into the roadmap, this method first puts the initial configuration into roadmap and then extends it with the set of sampled milestones. The reason is that in IPM the purpose is to escape the local minimum, so we have to first guarantee $\mathbf{u}_{loc}$ is in this roadmap. After $\mathbf{u}_{loc}$ is included in the roadmap, milestones that can be connected to $\mathbf{u}_{loc}$ are used to help the robotic sensor escape the local minimum. Since the roadmap is used for online sensor path planning and the robotic sensor is approximated to be a unicycle model, the local planer no longer uses a straight line between two milestones to check the existence of a path. For each milestone in the roadmap, a set T_i stores all the milestones that the robotic sensor will visit from c_0 to c_i. Meanwhile, W_i stores $[v_i^p \ w_i^p]^T$ which are the expected linear velocity and angular velocity when the robotic sensor visits c_i. Then when check whether a new milestone c_j can be connected to c_i, the local planer considers c_i as the initial configuration and c_j as the virtual goal, and a potential field based on (5.21) is utilized to predict

the path of the robotic sensor. If the robotic sensor is predicted to be able to arrive at c_j, then c_j is added to the roadmap and its predicted v_j^p and w_j^p are stored in W_j. The process of constructing the roadmap is shown in Fig. 5.4. The roadmap is initialized as $G_0 = c_0$. At the first step, the milestones, which can be connected to $\mathbf{u}_{loc}$ by the local planner, construct the first step roadmap G_1, and the remaining milestones form a set C_1 to construct the roadmap at the next step; then at step i, the milestones in C_{i-1} which can be connected to G_{i-1} are added to form G_i. The process stops when $G_{i-1} = G_i$, and then sets $G = G_i$. If $G = G_0$ the construction fails and new milestones need to be regenerated to construct the roadmap.

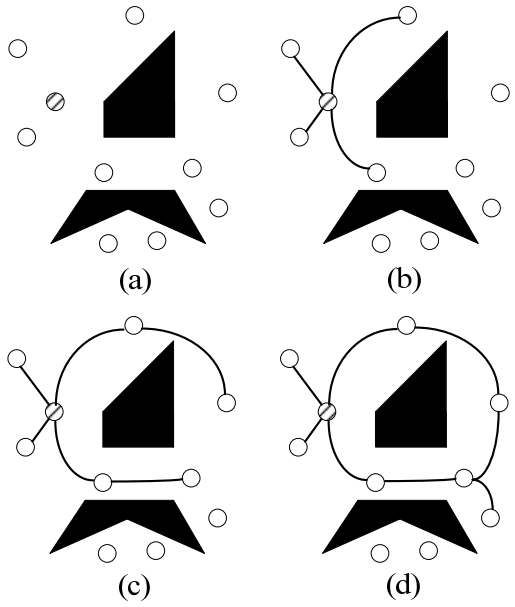


FIGURE 5.4: Process to construct the roadmap: (a) initial milestones; (b) first step; (c) second step; (d) final step. dash circle: local minimum; white circle: milestones; black area: C-obstacles

After the roadmap is constructed, it is utilized to move the robotic sensor away from the local minimum by checking whether a milestone in the roadmap can lead to a configuration with lower potential. The process is as follows:

Step I: A local information roadmap is constructed based on the local minimum at $\mathbf{u}_{loc}$ where the samples are generated with (5.25). Let the set $G_{left} = G$.

Step II: If G_{left} is empty, go to step IV; otherwise randomly choose a milestone $c_i \in G_{left}$ and predict the robotic sensor path starting from c_i based on the robotic sensor's potential field and achieve a new local minimum $\mathbf{u}'_{loc}$.

Step III: If $\mathbf{u}'_{loc} < \mathbf{u}_{loc}$, let $M = T_i$, and go to step V; otherwise, delete c_i from G_{left} and go back to step II.

Step IV: Randomly choose one set T_i from the roadmap, and let $M = T_i$.

Step V: Navigate the robotic sensor to visit all the milestones contained in M in order. If the robotic sensor gets stuck when visiting these milestones, it implies the workspace has changed. Then disregard the current roadmap and construct a new local PRM with this new local minimum.

In our simulation, we assume that, along the path, the robotic sensor can generate at most m local information roadmaps, where m is a constant parameter chosen by the user. If the robotic sensor fails to measure a target by generating no more than m local information roadmaps, then the robotic sensor changes to a temporary idle state and wait for a a new target assignment in the process. For a robotic sensor in exploration mode that is stuck by a local minimum, a similar strategy to the one proposed for the robotic sensor in the classification mode is used to construct its local roadmap. The difference lies in that the probability of generating a milestone are assumed to be equal around the robotic sensor in the exploration mode.

5.4 Complexity Analysis of Information Potential Method

In this section, some properties of the proposed method, such as complexity of target assignment problem, the expected time for reaching a target, and the average time to call the local planer in probabilistic roadmap construction, are analyzed and discussed.

5.4.1 Target Assignment Problem

Assume all the targets in T and robotic sensors in A can be represented as convex right prisms obeying the assumptions described in Section 5.1. The m targets are assigned to the r robotic sensors based on the distances between every robotic sensor and every target, where $n \geq 1$ and $r \geq 1$. Every detected target is assigned to one and only one robotic sensor. Since the distance between each pair of target and robotic sensor is computed, in the worst case the computational complexity of assigning the targets to the robotic sensors are $O(mr)$. It can be seen that the computational burden is high when the robotic sensor number and targets number are large. In this case, the entire workspace may be divided into a set of subspaces first, then targets are only assigned to robotic sensors in the same subspace.

5.4.2 Expected Time for Reaching a Target

Hereon we assume that the targets assigned to a robotic sensor are denoted by $T = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m\}$. A potential field is generated for the robotic sensor with the assigned targets and detected obstacles. Under proper assumptions, it can be shown that (1) starting at any initial configuration, the robotic sensor will converge to a C-target with finite number of calls to construct the local probabilistic roadmaps. (2) The target with a higher information value has a larger probability to be measured by the robotic sensor.

From the discussion of IPM, we know that all C-targets, $\mathcal{CT}_1, \mathcal{CT}_2, \ldots, \mathcal{CT}_m$, are global minima for the potential field. Furthermore, assume that there are finite local minima $\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_k$ rather than C-targets in the potential field. With the proposed control strategy as described in Section 5.5, starting at any $\mathbf{q} \in \mathcal{C}_{free}$, the robotic sensor will stop at a local minimum. If the local minimum is a C-target, the robotic sensor will stop moving and take measurement on the corresponding target. After that the target is classified and the left targets are assigned again to the robotic

sensors. If the local minimum is not a C-target, a local probabilistic roadmap is generated for the robotic senor in order to escape from the local minimum. Recall the PDF of the milestones generation

$$f(\mathbf{q}) = \begin{cases} \frac{e^{-U(\mathbf{q})}}{\int_{\mathcal{E}} e^{-U(\mathbf{q})} d\mathbf{q}} & \mathbf{q} \in \mathcal{E} \\ 0 & \mathbf{q} \notin \mathcal{E} \end{cases} \tag{5.26}$$

where $U(\mathbf{q})$ is the potential value at $\mathbf{q}$, and $\mathcal{E}$ is a randomly generated subspace around the local minimum. $\mathcal{E}$ may also be chosen as an area with a deterministic size by the user based on the size of potential field. It can be seen that $f(\mathbf{q})$ is always nonnegative.

The potential field can be partitioned into $(m + n)$ regions, $\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_m$, $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_n$ where starting at $\forall \mathbf{q} \in \mathcal{R}_i$, the robotic sensor will converge to $\mathcal{CT}_i$, and starting at $\forall \mathbf{q} \in \mathcal{L}_i$, the robotic sensor will converge to $\mathbf{l}_i$. For simplicity's sake, we represent the regions by $r_1, r_2, \ldots, r_m, t_1, t_2, \ldots, t_n$, where r_i corresponds to $\mathcal{R}_i$, and t_i corresponds to $\mathcal{L}_i$. Assume that the robotic sensor current configuration is $\mathbf{q} = \mathbf{l}_i$, then the probability to sample a milestone $\mathbf{s}$ in the region r_i and t_j is

$$p(\mathbf{s} \in r_i | \mathbf{q} = \mathbf{l}_i) = \int_{\mathbf{q}' \in \mathcal{R}_i} f(\mathbf{q}') d\mathbf{q}' \tag{5.27}$$

$$p(\mathbf{s} \in t_j | \mathbf{q} = \mathbf{l}_i) = \int_{\mathbf{q}' \in \mathcal{L}_j} f(\mathbf{q}') d\mathbf{q}' \tag{5.28}$$

Furthermore, assume that all milestones are connected to the roadmap, then at $\mathbf{l}_i$, $p(\mathbf{s} \in r_i | \mathbf{q} = \mathbf{l}_i)$ is the probability that the robotic sensor will travel from $\mathbf{l}_i$ to $\mathcal{CT}_j$. Denote this probability as $p(t_i, r_j) = p(\mathbf{s} \in r_i | \mathbf{q} = \mathbf{l}_i)$. If the jth target has higher information value, from previous discussion in Section 5.2, we know that $\mathcal{R}_j$ is larger, which provides a higher $p(t_i, r_j)$. Assume at step k, the robotic sensor is at $\mathbf{l}_i$, then $p(t_i, r_j)$ is independent to the local minimum that the robotic sensor stands in previous steps. Therefore, the movement of the robotic sensor can be modeled as

a Markov chain, similar to the case in [98]. The transition matrix can be presented as

$$P = \begin{pmatrix} p(r_1, r_1) & p(r_1, r_2) & \dots & p(r_1, t_1) & \dots, & p(r_1, t_n) \\ p(r_2, r_1) & p(r_2, r_2) & \dots & p(r_2, t_1) & \dots, & p(r_2, t_n) \\ & & & \dots & & \\ p(t_n, r_1) & p(t_n, r_2) & \dots & p(t_n, t_1) & \dots, & p(t_n, t_n) \end{pmatrix} \tag{5.29}$$

Write P as

$$P = \begin{pmatrix} I & R \\ S & Q \end{pmatrix} \tag{5.30}$$

where I is a $m \times m$ identity matrix, R is a $m \times n$ zero vector, S is a $n \times m$ matrix, and Q is a $n \times n$ matrix. Since Q is a matrix with nonnegative entries and the sum of its each row is not bigger than one, we have $\lim_{n\to\infty} Q^n \to 0$. Starting at t_i, denote the expected number of visiting t_j before the robotic sensor arrives at any r_k as $T(t_j|t_i)$, where $k = 1, 2, \dots, m$. $T(t_j|t_i)$ equals the ith row jth column of the matrix $C = (I - Q)^{-1}$ [99]. Then the expected steps for the robotic sensor to arrive at r_k, where $k = 1, 2, \dots, m$, starting at t_i is

$$T(t_i) = \sum_{j=1}^{n} c_{ij} \tag{5.31}$$

where c_{ij} is the ith row jth column of the matrix C. (5.31) shows that for every start position, the robotic sensor can converge to a target with finite calls of constructing the local probabilistic roadmap.

Now compute the probability for the robotic sensor to converge to r_j from t_i, denoted by $q(t_i, r_j)$.

$$q(t_i, r_j) = \sum_{x \in t_k, k=1,\dots,n} p(t_i, x) q(x, r_j) + p(t_i, r_j) \tag{5.32}$$

Write matrix A as the matrix with $a_{ij} = q(t_i, r_j)$, then from (5.32)

$$A = QA + S \tag{5.33}$$

which gives

$$A = (I - Q)^{-1}S = CS \tag{5.34}$$

from the definition of S, we know

$$S = \begin{pmatrix} p(t_1, r_1) & p(t_1, r_2) & \dots & p(t_1, r_m) \\ p(t_2, r_1) & p(t_2, r_2) & \dots & p(t_2, r_m) \\ & \dots & & \\ p(t_n, r_1) & p(t_n, r_2) & \dots & p(t_n, r_m) \end{pmatrix} \tag{5.35}$$

From previous discussion it can be seen that for a target j with higher information value than target h, $p(t_i, r_j)$ tends to be larger than $p(t_i, r_h)$ for each t_i. Then the probability

$$q(t_i, r_j) = \sum_{k=1}^{n} c_{ik} p(t_k, r_j) > \sum_{k=1}^{n} c_{ik} p(t_k, r_h) = q(t_i, r_h) \tag{5.36}$$

Therefore, the target with higher information value is more likely to be measured by the robotic sensor.

When $\mathcal{E}$ only covers a portion of the potential field around the robotic sensor's current configuration $\mathbf{l}_i$, some regions of C-target, r_j, that are outside $\mathcal{E}$ is impossible to be achieved at one step, i.e. $p(t_i, r_j) = 0$. In this case, not only the size of $\mathcal{R}_j$, but also the position of the target will effect the probability of the corresponding target to be measured by the robotic sensor. More specifically, the high informative targets with large number of local minima adjacent to it usually has high probability to be measured.

5.4.3 Computational Complexity of Local Probabilistic Roadmaps

When the local probabilistic roadmap is constructed, a local planer is utilized to check whether there exists a free path connecting a pair of milestones. Let the total number of calls of the local planer be t. If all pairs of milestones including the robotic sensor's initial configuration, are checked by the local planer, then $t = \frac{n(n+1)}{2}$, where

n is the number of milestones. However, in IPM, the average number to call the local planer is less. Assume at the ith step, the number of milestones added to the roadmap is m_i. Let q be the largest step that some milestones are added to the roadmap, we have $q \leq n$, $m_i \geq 0$, and $\sum_{i=1}^{q} m_i \leq n$. In the worst case,

$$t = \frac{n(n+1)}{2} - \sum_{i=1}^{q} C_{m_i}^2 - C_{n-\sum_{i=1}^{q} m_i}^2 \tag{5.37}$$

When all the milestones can be connected to the local minimum at the first step, i.e. $q = 1$ and $m_1 = n$, we have $t = n$ which gives a computational complexity $O(n)$. When $q = 1$ and $m_1 = kn$, where k is a positive constant less than one, $t = n + n^2(k - k^2)$ which give a computational complexity $O(n^2)$. Similarly for $q > 1$, the computational complexity is still $O(n^2)$. Although the worst computational complexity has the same order as the upper bound, the average number of t should be less than the upper bound. Assume that for each pair of milestones or a milestone and the local minimum, the probability for them to have a free path is p. Theoretic analysis for (5.37) is difficult, so we use an simulation example to show the average value of t for different values of p and n. All simulation results are averaged with 1000 runs. The results are shown in Figure 5.5 and Figure 5.6.

From the results we can see that for all chosen n, t achieves the highest value when p is less than 0.1. For $p > 0.2$ the highest value of t is about half of its corresponding upper bound and achieves at about $p = 0.5$. This shows that the time to call the local planer can be reduced at least 50% for most connected probabilities. Since the computational complexity is $O(n^2)$, for large n, t may also be too large. One way to further decrease t is to check only a certain number of milestones that are closest to the current node when call the local planer.

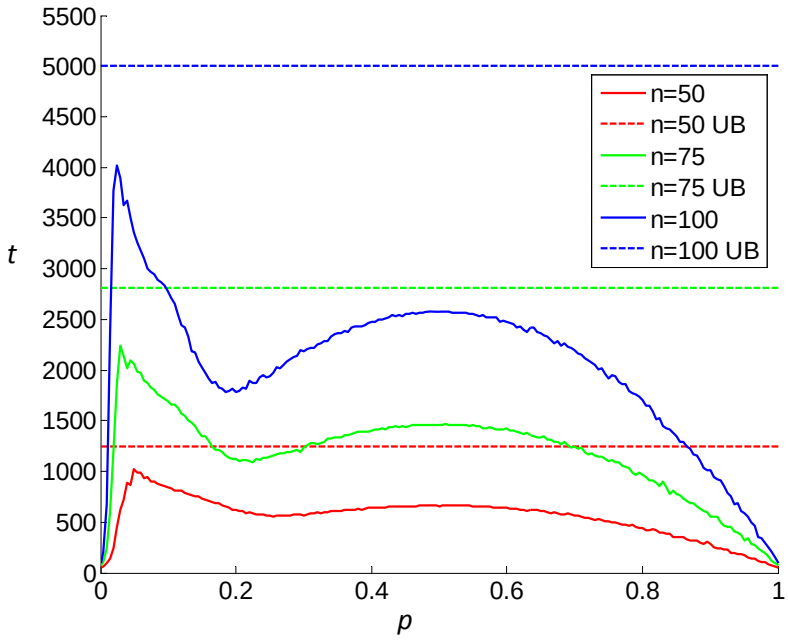


FIGURE 5.5: The average number to call the local planer when the connection probability changes under different n. The Upper Bound is $\frac{n(n+1)}{2}$ and is written as UB in the legend.

5.5 Switched Control of the Robotic Sensor

Since the robotic sensor is assumed to be a unicycle system, its control only has two dimensions which is less than the dimension of $\nabla U(\mathbf{q})$. Rewrite (5.1) in a matrix form

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} cos(\theta) \\ sin(\theta) \\ 0 \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} a \\ w \end{pmatrix} \tag{5.38}$$

Inspired by [96], a control algorithm is proposed to control the robotic sensor's movement. Assume that the potential field for the jth robotic sensor located at $\mathbf{q}$ is

$$U^j(\mathbf{q}) = U(\mathbf{q})_{rep} + U^j(\mathbf{q})_{att} + U^j_r(\mathbf{q}) \tag{5.39}$$

where

$$U^j(\mathbf{q})_{att} = \Pi_{i \in N(j)} U_i(\mathbf{q})_{att} \tag{5.40}$$

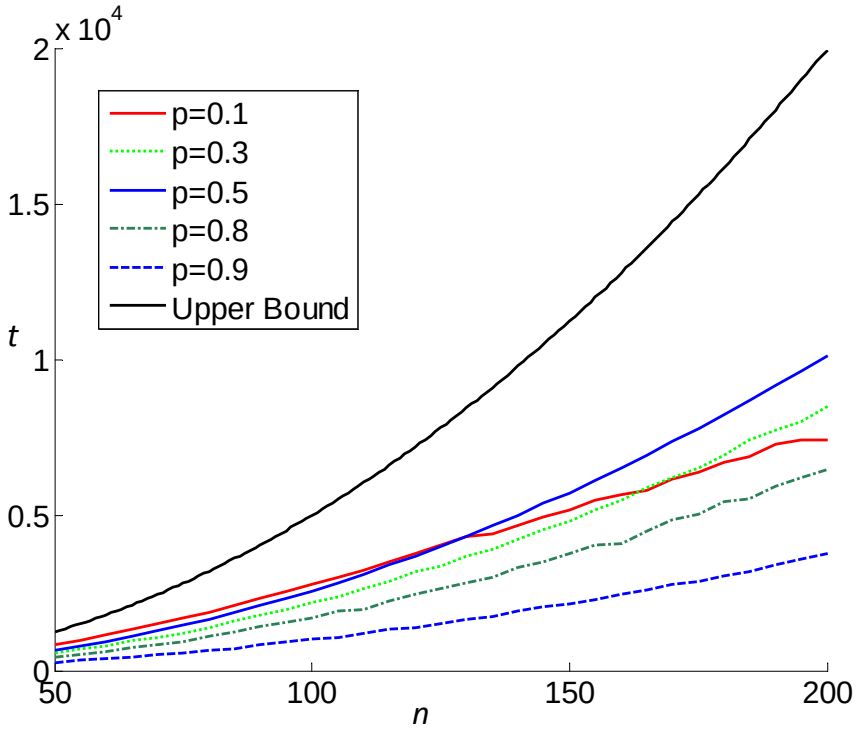


FIGURE 5.6: The average number to call the local planer when the number of milestones increases under different p. The Upper Bound is $\frac{n(n+1)}{2}$.

In order to achieve a stable control strategy, $\rho_i^t(\mathbf{q})$ is computed in the following way instead. Let $\mathbf{r} = [x\ y]^T$ be the position vector of $\mathbf{q}$ in the workspace. Before the targets are assigned to the robotic sensor network, a vector pointing to each target form the robotic sensor is computed by

$$\mathbf{h}_i = \lim_{\mathbf{h}} \{\mathbf{h} \mid \mathbf{h} = \|\mathbf{r}_2 - \mathbf{r}_1\|, \mathbf{r}_2 \in \mathcal{T}_i, \mathbf{r}_1 \in \mathcal{A}_j\} \qquad i = 1, 2, \ldots, m \tag{5.41}$$

And the orientation of target j to the robotic sensor, denoted by θ_f^i, is set by the heading of $\mathbf{h}_i$. Then the C-target associated with θ_f^i is computed and denoted by $\mathcal{CT}_i$. The position of $\mathcal{CT}_i$ to the robotic sensor is represented by its geometric center $\mathbf{r}_c^i = [x_c^i\ y_c^i]^T$. Let $\mathbf{q}_f^i = [x_c^i\ y_c^i\ \theta_f^i]^T$, $\rho_i^t(\mathbf{q})$ is computed by,

$$\rho_i^t(\mathbf{q}) = \|\mathbf{r}_c^i - \mathbf{r}\| \tag{5.42}$$

where $\|\cdot\|$ represents the Euclidian norm.

The width of the $\mathcal{CT}_i$ is described by the radius of the biggest circle centered at $\mathbf{r}_c^i$ and contained by the closure of $\mathcal{CT}_i$ denoted as r_i and shown in Fig. 5.7.

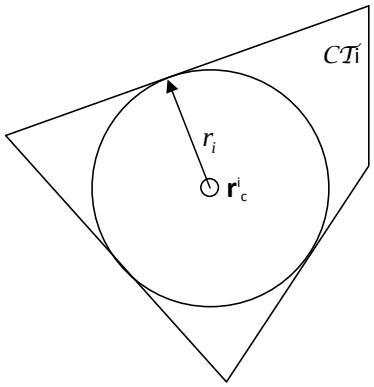


FIGURE 5.7: The inscribed circle of the ith C-target and its center $\mathbf{r}_c^i$ and radius r_i

The control of the robotic sensor is divided into two steps based on the robotic sensor's distance to its closest C-target. Let $\mathbf{h} = \lim_i \mathbf{h}_i$, and $\epsilon < r_i$ be a positive constant. When $\mathbf{h} > \epsilon$, θ_f is not considered in the control, in which case, the goal is a vertical line in the three-dimensional configuration space with the length 2π, denoted by $\mathbf{Q}_f$, and the set of configurations with distance no more than a specified distance ϵ to $\mathbf{Q}_f$ is a cylinder as shown in Fig. 5.8. The robotic sensor is first navigated into the cylinder, and then θ_f is considered to change the robotic sensor's orientation.

Let

$$S(\mathbf{q}) = \begin{pmatrix} cos(\theta) \\ sin(\theta) \\ 0 \end{pmatrix} \tag{5.43}$$

When $\mathbf{h} > \epsilon$, define the control law as

$$u_1 = -S(\mathbf{q})^T \nabla U(\mathbf{q}) - k_1 v \tag{5.44}$$

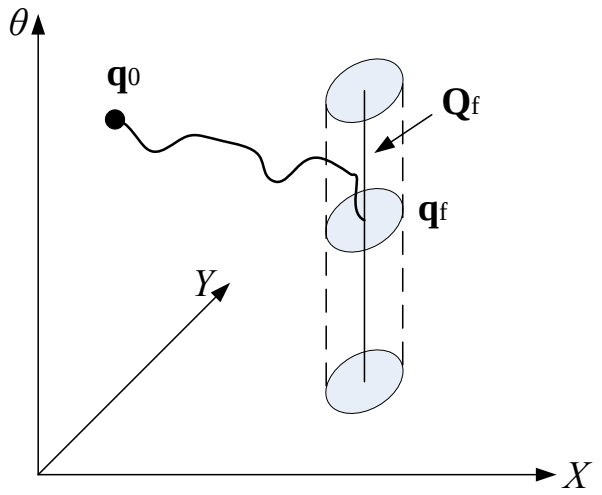


FIGURE 5.8: An example path from $\mathbf{q}_0$ to $\mathbf{q}_f$

where k_1 is a constant parameter.

$$u_2 = \dot{\alpha}(U(\mathbf{q})) + k_0(\alpha(U(\mathbf{q})) - \theta) \tag{5.45}$$

where $\alpha(U(\mathbf{q}))$ is the orientation angle of the vector $[\frac{\partial U(\mathbf{q})}{\partial x}, \frac{\partial U(\mathbf{q})}{\partial y}]^T$, and k_0 is a positive constant. When $U(\mathbf{q})$ is second differentiable, and $\frac{\partial U(\mathbf{q})}{\partial x} \neq 0$, $\alpha(U(\mathbf{q}))$ can be computed as

$$\alpha(U(\mathbf{q})) = \arctan\left(\frac{\frac{\partial U(\mathbf{q})}{\partial y}}{\frac{\partial U(\mathbf{q})}{\partial x}}\right) \tag{5.46}$$

$$\begin{aligned}\dot{\alpha}(U(\mathbf{q})) =& \frac{\frac{\partial U(\mathbf{q})}{\partial x}}{(\frac{\partial U(\mathbf{q})}{\partial x})^2 + (\frac{\partial U(\mathbf{q})}{\partial y})^2}(\frac{\partial^2 U(\mathbf{q})}{\partial x \partial y}\dot{x} + \frac{\partial^2 U(\mathbf{q})}{\partial y^2}\dot{y}) \\ & - \frac{\frac{\partial U(\mathbf{q})}{\partial y}}{(\frac{\partial U(\mathbf{q})}{\partial x})^2 + (\frac{\partial U(\mathbf{q})}{\partial y})^2}(\frac{\partial^2 U(\mathbf{q})}{\partial x \partial y}\dot{y} + \frac{\partial^2 U(\mathbf{q})}{\partial x^2}\dot{x})\end{aligned} \tag{5.47}$$

After the robotic sensor is within ϵ distance to a C-target's center, the control law of the system changes to

$$u_1 = -K_p S(\mathbf{q})^T \nabla U(\mathbf{q}) - k_1 v \tag{5.48}$$

where k_1 and K_p are constant parameters.

$$u_2 = k_0(\theta_f - \theta) \tag{5.49}$$

The stability of the control algorithm can be proved under the following assumption. (i) The robotic sensor is approaching one target and its movement is dominated by this C-target, i.e. the distance of the robotic sensor to other C-targets is big enough to eliminate other targets' force. Without loss of generality, we assume the robotic sensor's movement is governed by $\mathcal{CT}_1$. (ii) There is no other robotic sensor in the influence distance of the robotic sensor. (iii) The robotic sensor is not in the influence distance of any obstacle. Based on assumption (ii) in (5.39) $U_r^j(\mathbf{q}) = 0$. Then (5.17) is written as

$$\nabla U(\mathbf{q}) = \sum_{i \in I_{\rho_0}} M_i^B \mathbf{n}_i^B + \sum_{i=1}^{m} (N_i + M_i^T)\mathbf{n}_i^T \tag{5.50}$$

With assumption (iii), $I_{\rho_0} = \emptyset$ and $M_i^T = 0$ for all i. Finally with assumption (i), $N_i = 0$ for all $i \neq 1$. Therefore (5.50) is simplified to

$$\begin{aligned} \nabla U(\mathbf{q}) &= N_1 \mathbf{n}_i^T \\ &= \Pi_{j=2}^{m} U_j(\mathbf{q})_{att} \eta_2 e^{-\frac{\rho_i^t(\mathbf{q})^2}{2\sigma V_i^a}} \rho_i^t(\mathbf{q}) \nabla \rho_1^t(\mathbf{q}) \end{aligned} \tag{5.51}$$

When $\mathbf{q}$ is outside the cylinder with a radius $0 < \epsilon < r$ as shown in Fig. 5.8, $\rho_1^t(\mathbf{q})$ is replaced as the distance between $\mathbf{q}$ and $\mathbf{Q}_f$. Assume $\mathbf{q} = [x\ y\ \theta]^T$ and the position of $\mathbf{Q}_f$ is $[x_f\ y_f]^T$, we have

$$\rho_1^t(\mathbf{q}) = \|(x_f, y_f) - (x, y))\| \tag{5.52}$$

$$\nabla \rho_1^t(\mathbf{q}) = \begin{pmatrix} \frac{(x_f - x)}{\rho_1^t(\mathbf{q})} & \frac{(y_f - y)}{\rho_1^t(\mathbf{q})} & 0 \end{pmatrix}^T \tag{5.53}$$

Substitute into (5.51)

$$\nabla U(\mathbf{q}) = \begin{pmatrix} -K(x_f - x) & -K(y_f - y) & 0 \end{pmatrix}^T \tag{5.54}$$

where $K = \Pi_{j=2}^{m} U_j(\mathbf{q})_{att} \eta_2 e^{-\frac{\rho_1^t(\mathbf{q})^2}{2\sigma V_1^a}}$.

The Lyapunov function candidate is

$$V = U(\mathbf{q}) + \frac{1}{2}v^2 + \frac{1}{2}(\alpha(U(\mathbf{q})) - \theta)^2 \tag{5.55}$$

Then

$$\begin{aligned}
\dot{V} &= \nabla U(\mathbf{q})^T \dot{\mathbf{q}} + v\dot{v} + (\alpha(U(\mathbf{q})) - \theta)(\dot{\alpha}(U(\mathbf{q})) - w) \\
&= \nabla U(\mathbf{q})^T \begin{pmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} + v\left(-S(\mathbf{q})^T \nabla U(\mathbf{q}) - k_1 v\right) - k_0 \left(\alpha(U(\mathbf{q})) - \theta\right)^2 \\
&= -k_1 v^2 - k_0(\alpha(U(\mathbf{q})) - \theta)^2 \leq 0
\end{aligned} \tag{5.56}$$

Since $U(\mathbf{q})$ is bounded below, the LaSalle's invariance principle [100] implies that the system converges to $v = 0$ and $\theta = \alpha(U(\mathbf{q}))$. Furthermore, when $v = 0$, the control u_1 which is the acceleration of the robotic sensor also needs to be zero; otherwise, the non-zero input will change v to a non-zero value. Then from (5.44) we have

$$\frac{\partial U(\mathbf{q})}{\partial x} cos(\theta) + \frac{\partial U(\mathbf{q})}{\partial y} sin(\theta) = 0 \tag{5.57}$$

Since $\theta = \alpha(U(\mathbf{q}))$, we have $\frac{\partial U(\mathbf{q})}{\partial x} = \|\nabla U(\mathbf{q})\| cos(\theta)$ and $\frac{\partial U(\mathbf{q})}{\partial y} = \|\nabla U(\mathbf{q})\| sin(\theta)$, then from (5.57)

$$\|\nabla U(\mathbf{q})\| cos(\theta)^2 + \|\nabla U(\mathbf{q})\| sin(\theta)^2 = 0 \tag{5.58}$$

which results in $\nabla U(\mathbf{q}) = 0$. From (5.54), we have

$$\begin{aligned}
x &= x_f \\
y &= y_f
\end{aligned} \tag{5.59}$$

Since the robotic sensor asymptotically converges to (x_f, y_f), there is no guarantee that the robotic sensor can achieve the goal in finite time. Therefore, this robotic

control law switches to the ones in (5.48) and (5.49) when $\|(x, y) - (x_f, y_f)\| < \epsilon$. This is guaranteed to be achieved in finite time [95].

When $\|(x, y) - (x_f, y_f)\| < \epsilon$, the Lyapunov function candidate is

$$V = K_p U(\mathbf{q}) + \frac{1}{2}v^2 + \frac{1}{2}(\theta_f - \theta)^2 \tag{5.60}$$

$$\dot{V} = -k_1 v^2 - k_0(\theta_f - \theta)^2 \leq 0 \tag{5.61}$$

which shows that θ converges to θ_f, and v converges to zero. Furthermore, since the potential function with $\|(x, y) - (x_f, y_f)\| < r$ is an increase function to $\rho_1^t(\mathbf{q})$, assume the infimum of the potential function with $\rho_1^t(\mathbf{q}) = r$ is bounded below by a constant M, then

$$K_p U(\mathbf{q}) + \frac{1}{2}v^2 + \frac{1}{2}(\theta_f - \theta)^2 \leq K_p U(\mathbf{q}) + \frac{1}{2}(v_{max}^2 + 4\pi^2) < MK_p \tag{5.62}$$

If we set

$$K_p > \frac{v_{max}^2 + 4\pi^2}{2(M - U(\mathbf{q}))} \tag{5.63}$$

the movement of the robotic sensor is constrained into the cylinder that has distance to (x_f, y_f) less than r. Then when $\theta = \theta_f$ and $v = 0$, the distance between the robotic sensor and $\mathbf{q}_f$ is less than r, i.e. the robotic sensor is within $\mathcal{CT}_1$, and the target can be measured by the robotic sensor.

5.6 Simulations and Results

In this section, the information potential method is tested in a demining system and workspace similar to Sections 4.4.2 and 4.5 with the following change.

(i) The workspace is in three-dimensional space. The robotic sensor platform is assumed to be a right prism. The geometry of the GPR sensor FOV is assumed as a cone, and the targets and obstacles are assumed to be right prisms as discussed in Section 5.1.

(ii) The IR measurement of a target can be either available *a priori* from a global sensor or needs to be measured by an IR sensor mounted on the robotic sensor.

(iii)In the workspace it is assumed to have N targets deployed with their prior IR measurements either known or unknown *a priori*. The robotic sensor network is asked to measured M targets with the onboard GPR sensor.

In Section 5.6.1, simple examples are given to explain how the IPM moves the robotic sensor in the workspace. In Section 5.6.2 the efficiency of the IPM is compared to that of the other sensor path planning methods such as rapidly-exploring random trees [101] and the classical potential method without taking the target information value into account when constructing the potential field.

5.6.1 Properties of the Information Potential Method

In this section, a series of simple examples are used to demonstrate why the prior measurements on targets and obstacles need to be taken into account in planning the sensor path. The examples also show how the potential field escape the local minimum and the efficiency of utilizing multiple robotic sensors. Fig. 5.9 and Fig. 5.10 illustrate that the effect of information value on the value of potential function. The workspace contains one obstacle (black) and two unequally-important target, $\mathcal{T}_1$ with information value 0.2 and $\mathcal{T}_2$ with information value 0.1. In this example $M = 1$. Although the two targets are symmetrically deployed above and below the obstacle and have the same geometry, the potential field tends to have a larger area with low potential value around $\mathcal{T}_1$ since it has a bigger information value. This example also shows that the IPM can successfully escape from the local minimum and measure the target.

Besides the information value of targets, the information potential method also

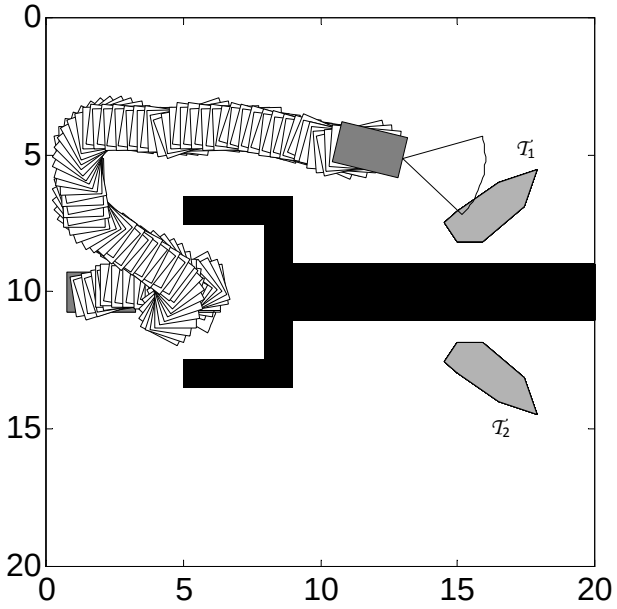


FIGURE 5.9: An example path generated by the information potential method. Target information value $V_1 = 0.2$, $V_2 = 0.1$.

takes into account the online obtained information such as new detected targets and obstacles. In Fig. 5.11 four targets are deployed in the workspace with $\mathcal{T}_1$, $\mathcal{T}_2$ and $\mathcal{T}_4$ undetected *a priori* (gray area with dash boundary). In this example, besides the GPR sensor, an IR sensor is also mounted on the robotic sensor (half circle with blue boundary) to detect new targets, M is set to be 2. It can be seen that at the beginning the robotic sensor moves toward the detected sensor $\mathcal{T}_3$. As soon as it detects $\mathcal{T}_2$, the high information value from $\mathcal{T}_2$ attracts the robotic sensor to change its direction and move toward $\mathcal{T}_2$. After $\mathcal{T}_2$ is measured by the GPR sensor, the robotic sensor continues to move to measure another new detected high value target $\mathcal{T}_1$. In Fig. 5.12 seven obstacles are deployed in the workspace with $\mathcal{B}_1$ and $\mathcal{B}_5$ undetected *a priori* (dark gray area with dash boundary). An IR sensor is also mounted on the robotic sensor to detect the obstacles and in this example $M = 1$. From the results we can see that the robotic sensor first moves toward $\mathcal{T}_1$, however,

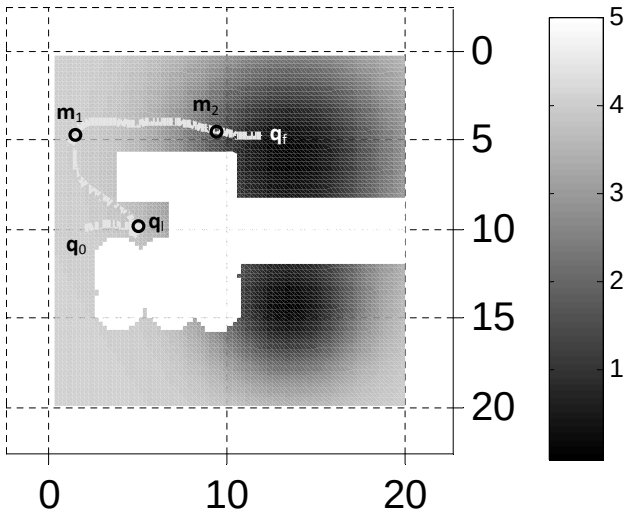


FIGURE 5.10: The potential field when $\theta = \frac{3\pi}{2}$. $\mathbf{q}_0$: initial configuration; $\mathbf{q}_f$: final configuration; $\mathbf{q}_l$: local minimum; $\mathbf{m}_1$, $\mathbf{m}_2$: milestones in the local PRM chosen by the robotic sensor (The potential value is pruned above at 5).

when $\mathcal{B}_5$ is detected by the robotic sensor, its repulsive potential prevents the robotic sensor moving toward $\mathcal{T}_1$. The robotic sensor then changes its direction and moves toward $\mathcal{T}_2$.

The following example shows why the target height is taken into account in the sensor path planning. In Fig. 5.13 the robotic sensor is asked to measure one target in the workspace. Target $\mathcal{T}_1$ and $\mathcal{T}_2$ have the same geometry and information value, and are deployed into the workspace at symmetrical position to the robotic sensor's initial configuration. However, the robotic sensor chooses to move toward $\mathcal{T}_1$ which has a lower height. The reason is that the difference of target height affects the shape and location of the two-dimensional sensor FOV for each target. As a result, $\mathcal{T}_1$ has a smaller distance to the robotic sensor although the two targets' positions on the

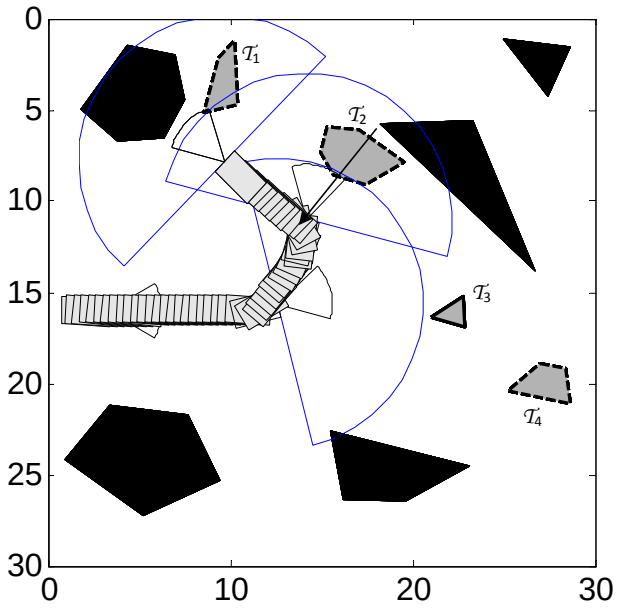


FIGURE 5.11: The path generated with a single robotic sensor to illustrate the influence of online target detection. Target information value $V_1 = 0.2$, $V_2 = 0.2$, $V_3 = 0.1$, $V_4 = 0.1$.

horizontal plane are the same.

Since to have small travel distance is one of the objective for the sensor path planing problem, the following example shows how multiple robotic sensors are utilized in IPM to reduce the travel distance by the robotic sensor network. From the example in Fig. 5.14 and Fig. 5.15 we can see how IPM plans the sensor path for both a single robotic sensor and a sensor network including two robotic sensors. The total travel distance of the sensor network decreased to 33.69 comparing to 43.50 of the single robotic sensor.

In previous section, the stability of the control strategy on the robotic sensor is proven under certain assumptions. However, there is no assumption on the bound of the linear acceleration a, linear velocity v, and angular velocity w. The following

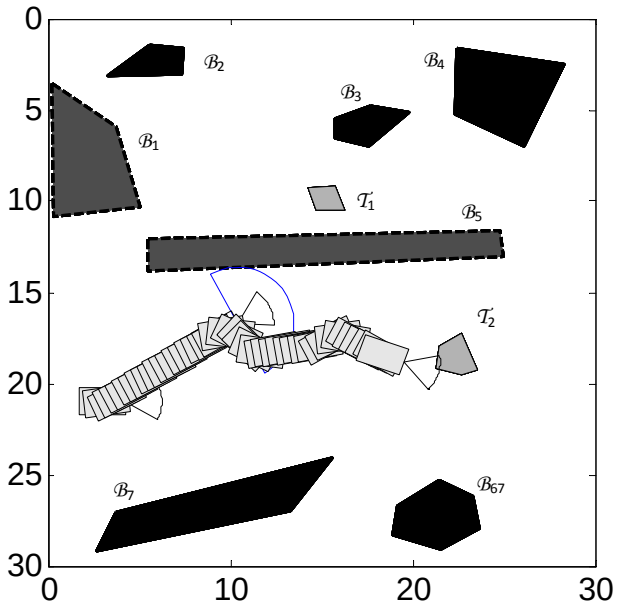


FIGURE 5.12: The path generated with a single robotic sensor to illustrate the influence of online obstacle detection. Target information value $V_1 = 0.2$, $V_2 = 0.2$.

example shows that the control of the robotic sensor may still be stable when there is a bound on a, v and w. In Fig. 5.16 the robotic sensor is asked to measure one target in the workspace. The robotic sensor's movement has these constraints: $|a| < 5m/s^2$, $|v| < 2m/s$, and $|w| < \frac{\pi}{10} rad/s^2$. The goal of the robotic sensor (x_f, y_f) is marked as a star in Fig. 5.17 while the final position of the robotic sensor (x_r, y_r) is marked as a cross. Together with Fig. 5.18, Fig. 5.19, and Fig. 5.20 we can see that the robotic sensor converges to a position within the inscribed circle of the C-target and has the same heading as the goal.

5.6.2 Efficiency Comparison and Results

This section summary the results of the information potential field method on a large minefield. The workspace is shown in Fig. 5.21. Three robotic sensors are utilized in this workspace to measure $M = 7$ targets. Three methods, information potential

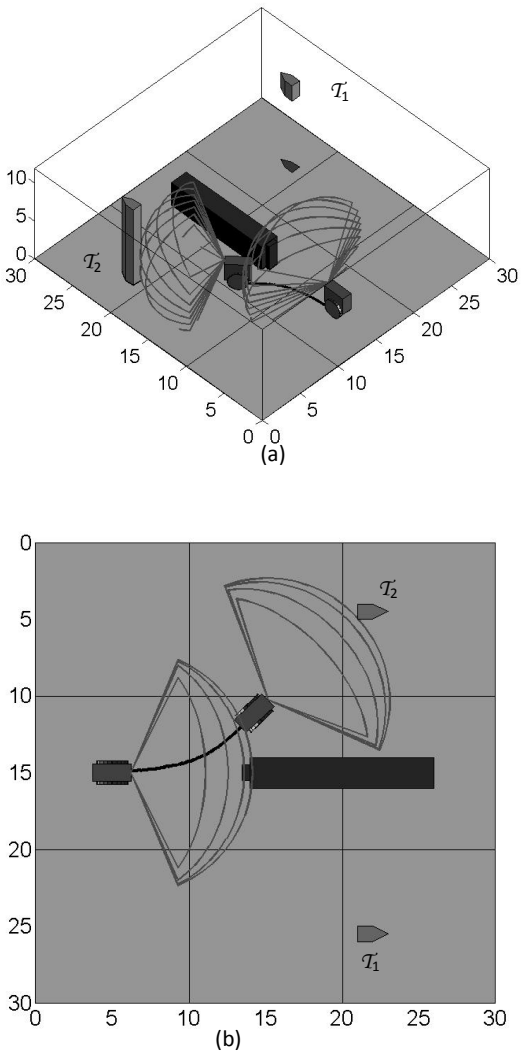


FIGURE 5.13: The path generated with a single robotic sensor to illustrate the influence of target height. Target information value $V_1 = 0.2$, $V_2 = 0.2$.

method, rapidly-exploring random trees [101], and a classical potential field that does not take into account the target information value are used to plan the robotic sensors' paths.

Similar to Chapter 4, the robotic sensor efficiency is computed as

$$\eta = \frac{N_{IRGPR} - N_{IR}}{D} \times 100\% \tag{5.64}$$

where N_{IRGPR} is the number of correctly classified targets with the fused measure-

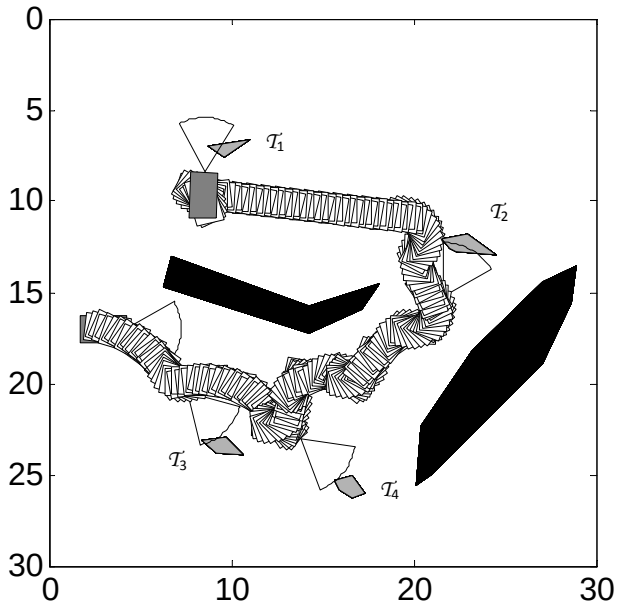


FIGURE 5.14: The path generated with a single robotic sensor to illustrate the influence of robotic sensor number. Target information value $V_1 = 0.15$, $V_2 = 0.15$, $V_3 = 0.15$, $V_4 = 0.15$.

ments from GPR sensor and IR sensor, N_{IR} is the number of correctly classified targets with only IR sensor measurement, and D is the total travel distance by the robotic sensor network. The results of these methods are shown in Table 5.1. From the results we can see that the IPM outperforms the other two methods both when IR measurements on obstacles and targets are available *a priori* or not. One interesting outcome is that the classical potential field method performs slightly better than the RRT method. The reason is that when prior information on some targets becomes available, the potential field method can help the robotic sensor avoid unnecessary path in exploring the workspace while RRT method still needs to sample in the workspace which may cause extra travel distance. An example path generated by the information potential method is shown in Fig. 5.22, and that by RRT is shown in 5.23. From the results we can see that when the prior information on targets is

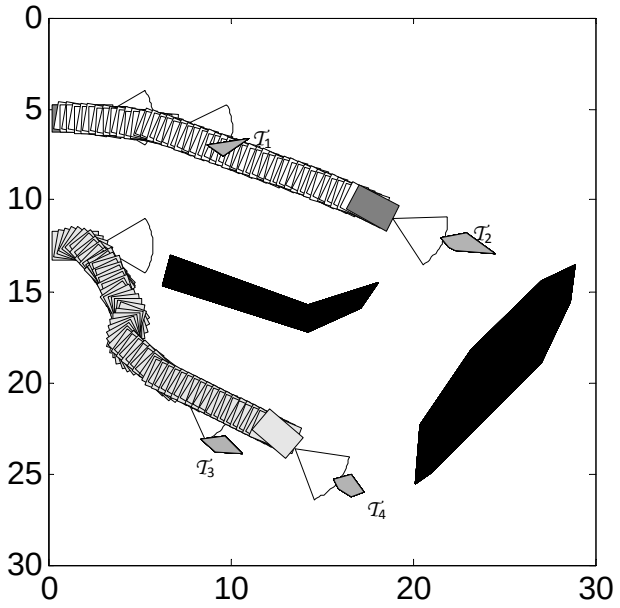


FIGURE 5.15: The path generated with two robotic sensors to illustrate the influence of robotic sensor number. Target information value $V_1 = 0.15$, $V_2 = 0.15$, $V_3 = 0.15$, $V_4 = 0.15$.

available, the IPM tends to measure the target with higher information value. For example, robotic sensors measure $\mathcal{T}_6$ instead of $\mathcal{T}_{10}$, and $\mathcal{T}_{12}$ instead of $\mathcal{T}_5$ when each pair of targets have similar distance to the assigned robotic sensor. As a sampled based method with sensor dynamics considered, the path generated by RRT tends to be very curved. For classical potential method path as shown in Fig 5.24, the first robotic sensor measures $\mathcal{T}_5$ instead of $\mathcal{T}_{12}$ since $\mathcal{T}_5$ has a smaller distance to $\mathcal{T}_4$.

Another workspace as shown in Fig. 5.25 is also utilized to test how the IPM takes advantage of the online target information. The workspace includes 18 targets and 9 obstacles, and four robotic sensors are required to measure $M = 10$ targets with its onboard GPR sensor. Some of the properties of the IPM such as online

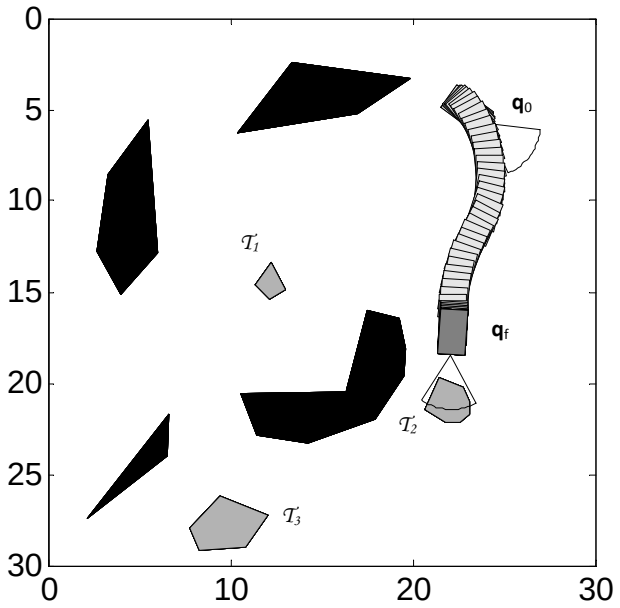


FIGURE 5.16: The path generated with a robotic sensor. Target information value $V_1 = 0.2$, $V_2 = 0.2$, $V_3 = 0.2$.

update of target assignment or utilization of target information value are tested. The results are included in Table 5.2.

From the results we can see that all the properties play an important role in the efficiency of IPM. Turn off any of the property may cause a decrease on the path efficiency.

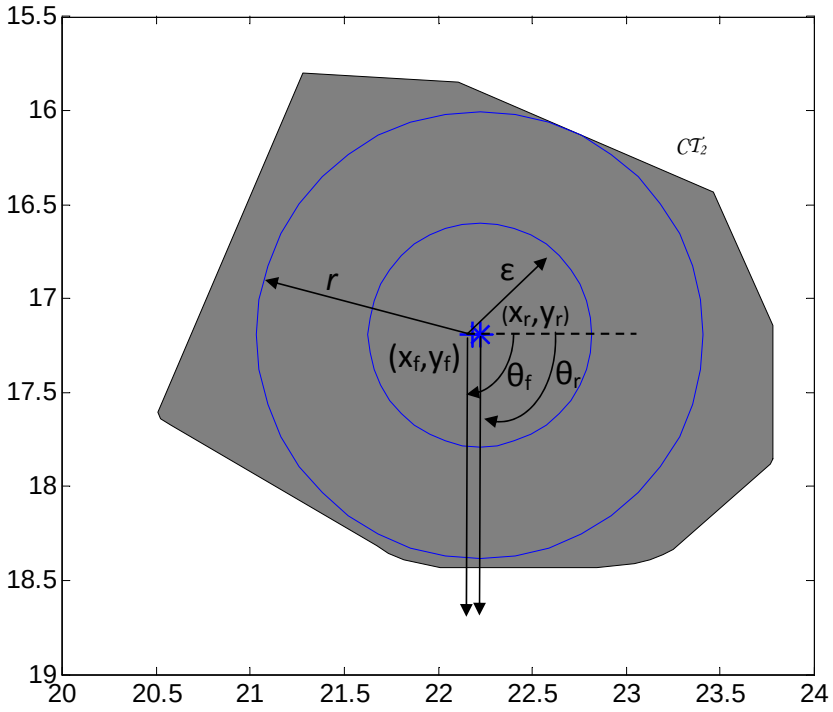


FIGURE 5.17: The final configuration and the goal configuration of the robotic sensor.

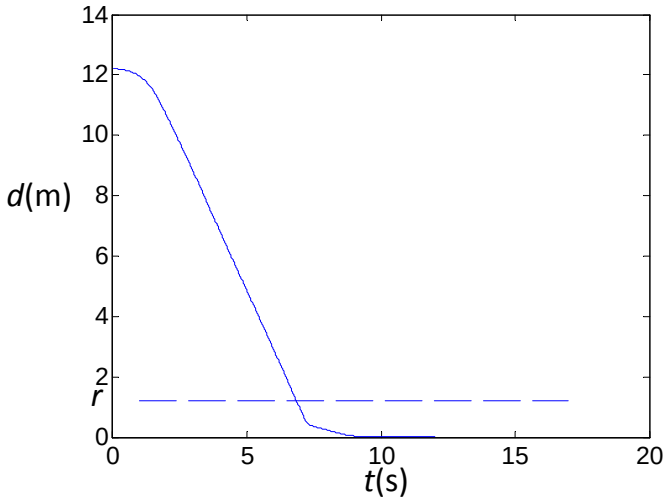


FIGURE 5.18: The change of distance between the robotic sensor and the goal.

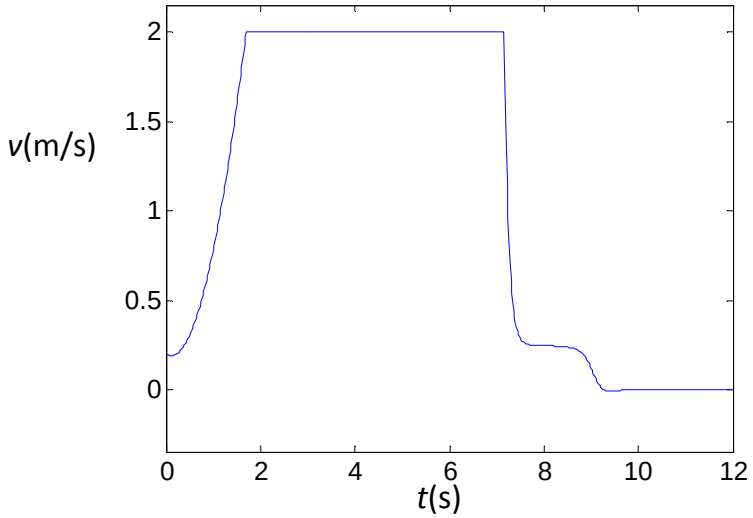


FIGURE 5.19: The change of linear velocity of the robotic sensor.

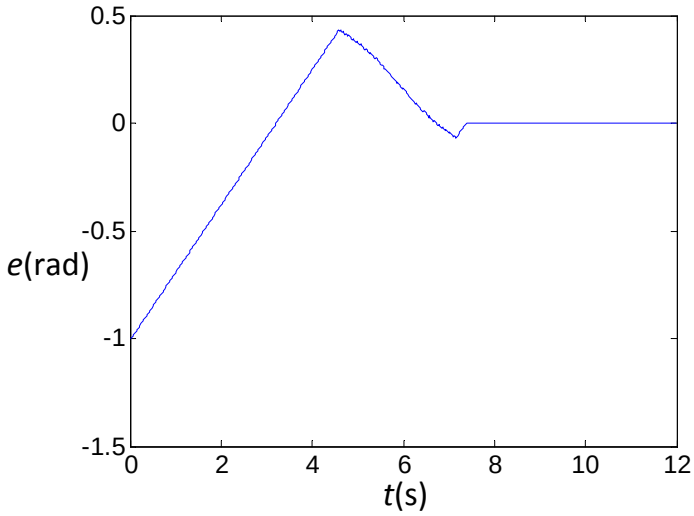


FIGURE 5.20: The change of difference between the robotic sensor heading and the goal orientation.

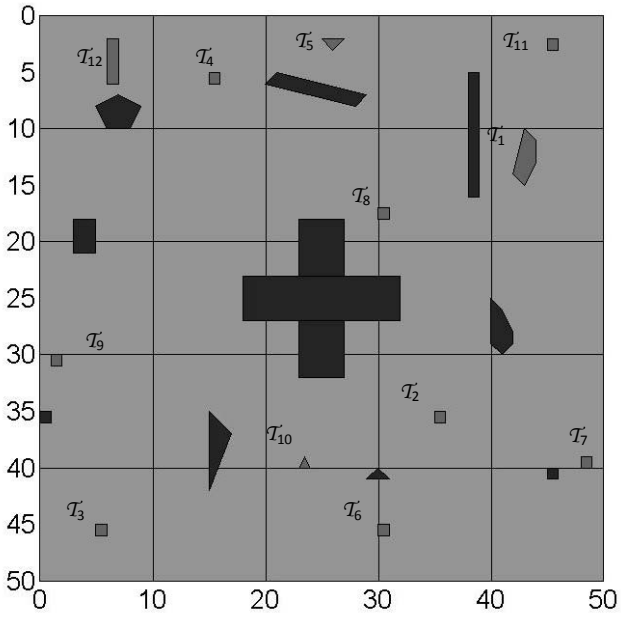


FIGURE 5.21: The workspace with 12 targets and 10 obstacles. Targets with information value bigger than 0.15: $\mathcal{T}_3$, $\mathcal{T}_4$, $\mathcal{T}_6$, $\mathcal{T}_7$, $\mathcal{T}_9$, $\mathcal{T}_{10}$, $\mathcal{T}_{12}$; Targets with information value between 0.10 and 0.15: $\mathcal{T}_1$, $\mathcal{T}_2$, $\mathcal{T}_8$; Targets with information value less than 0.10: $\mathcal{T}_5$, $\mathcal{T}_{10}$.

Table 5.1: The efficiency of three methods under different conditions. The results are averaged by 30 runs.

Method	Performance	
	Prior measurements on all obstacles and targets are available	No prior measurement on target or obstacle is available
IPM	0.0622	0.0333
RRT	0.0187	0.0142
Classical potential method	0.0209	0.0160

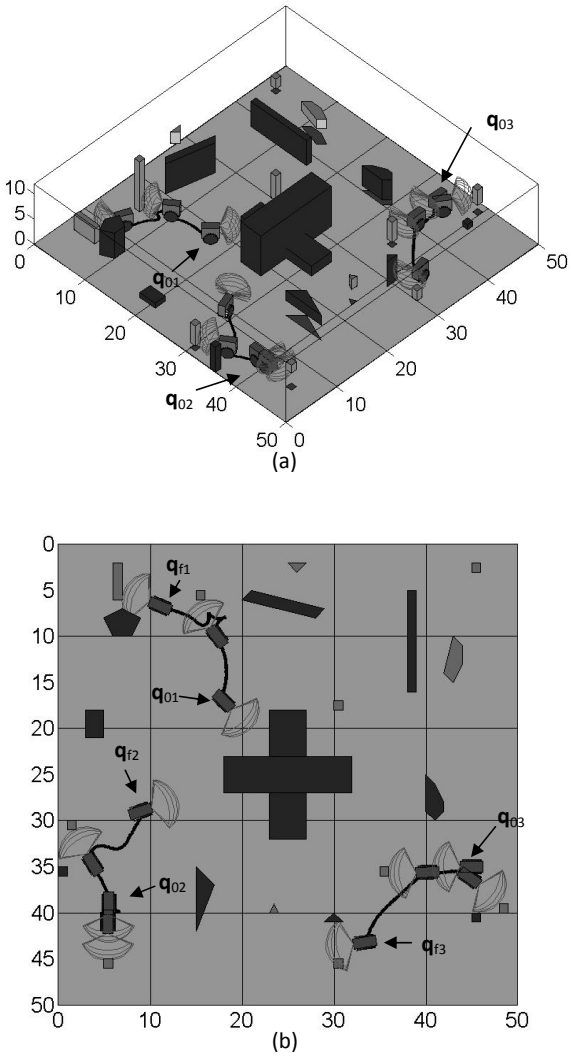


FIGURE 5.22: A path generated by the information potential method.

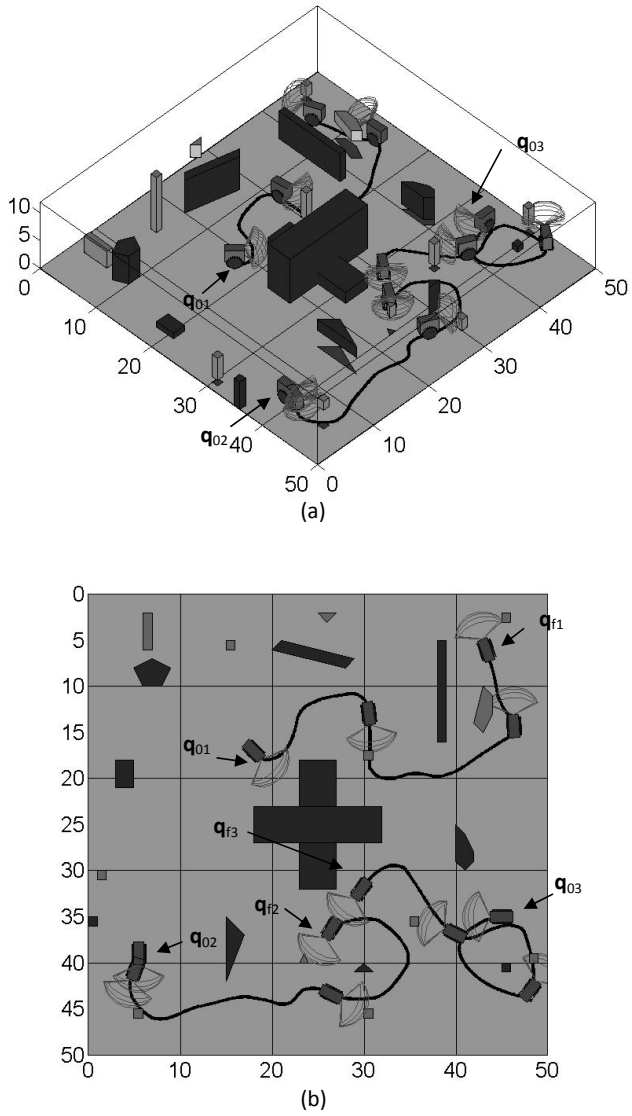


FIGURE 5.23: A path generated by the rapidly-exploring random trees method.

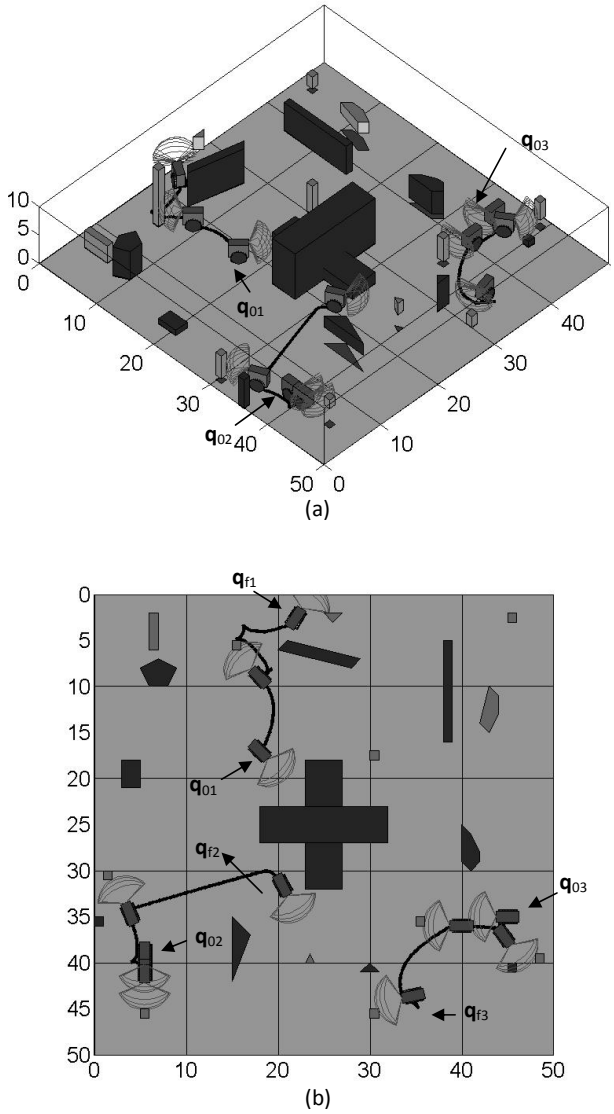


FIGURE 5.24: A path generated by the classical potential method.

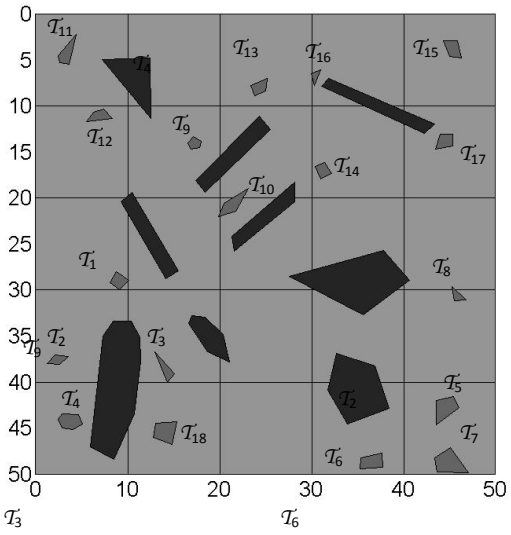


FIGURE 5.25: The workspace with 18 targets and 9 obstacles. Targets with information value bigger than 0.15: $\mathcal{T}_1$, $\mathcal{T}_2$, $\mathcal{T}_4$, $\mathcal{T}_5$, $\mathcal{T}_6$, $\mathcal{T}_7$, $\mathcal{T}_8$, $\mathcal{T}_{10}$, $\mathcal{T}_{13}$, $\mathcal{T}_{15}$; Targets with information value between 0.10 and 0.15: $\mathcal{T}_9$, $\mathcal{T}_{12}$, $\mathcal{T}_{14}$, $\mathcal{T}_{17}$; Targets with information value less than 0.10: $\mathcal{T}_3$, $\mathcal{T}_{11}$, $\mathcal{T}_{16}$, $\mathcal{T}_{18}$.

Table 5.2: The efficiency of IPM with different properties turned off. The results are averaged by 30 runs (The result of RRT is also included for comparison).

Condition	Efficiency
All properties included	0.0898
Turn off online target assignment	0.0475
Turn off target assignment all the time	0.0099
Turn off utilization of target information, as well as target and obstacle geometries obtained *a priori* (Equivalent to the case that all targets and obstacles are unknown *a priori*)	0.0346
Turn off utilization of target information all the time (Equivalent to the method of classical potential method)	0.0470
Single robotic sensor	0.0088
RRT	0.0235

6

Conclusions

The objective of the information-driven robotic sensor path planning is to plan the path for gathering sensor measurements for a sensing objective. This dissertation presents a general and systematic approach for deriving information value functions that represent the expected utility of the sensor decisions in a canonical sensor planning problem. The resulting information functions can then be implemented by an IS strategy that selects the measurement sequence with the highest expected information value. The IS strategy is compared to other search strategies previously presented in the literature by means of extensive numerical simulations involving several sensor models, priors, and decision rules. Consequently, the comparative performance results can be generalized beyond a particular sensor type or application.

A novel IRM approach that combines information theory with probabilistic roadmap methods is then presented and implemented in a simulated demining system. IRM computes a robotic sensor path that accounts for the geometry of the sensor's platform and field of view, and for the geometric characteristics of a workspace that is populated with multiple fixed targets and obstacles. The novel learning- and

query-phase algorithms use the targets' information value to generate a roadmap with a high density of high information-value milestones, and a path that optimizes a desired trade-off between sensing performance and distance traveled. The value of information is quantified by the expected entropy reduction metric. By implementing an existing BN approach to model the sensor measurements, the expected entropy reduction can be computed from the BN CPTs and from prior information, such as, prior sensor measurements and environmental conditions. The IRM method is demonstrated by planning the path of a GPR sensor installed on a ground robot, based on prior measurements obtained by an airborne IR sensor, and environmental information, such as, soil characteristics, vegetation, and weather (Section 4.4). The results, obtained from the simulations in Section 4.5, show that IRM outperforms existing sensor path-planning methods applicable to geometric sensing, such as, complete coverage and random search, under a wide range of workspace conditions and geometric characteristics, increasing the average path efficiency by up to one order of magnitude. Also, the novel learning phase presented in Section 4.2 can be used to plan the path of non-overpass capable robotic sensors that can be seriously damaged or even destroyed when driving over landmines. As a result, IRM displays a classification efficiency several times greater than that of other methods, and the robotic sensor can make measurements even from targets located inside narrow passages which, in this case, further restrict the configuration space, as they constitute potential obstacles.

When information of the workspace such as targets and obstacles geometries, and cursory measurements on targets, may become available online, an novel information potential method is proposed to take into account this online information in potential field construction for multiple sensor path planning. The targets are assigned to the robotic sensors based on the distances between every target and every robot. And a switch control strategy is integrated into the sensor path planning problem to control

the robotic sensor in the workspace. The potential function is also used to generate a local PRM to help the robotic sensor escape its local minimum. Experiments show that paths obtained from the information potential method takes advantages of the online information and coordination among robotic sensors, and the results show that the IPM outperforms other strategies such as rapidly-exploring random trees and classical potential field method without considering the information value.

Future work are recommended as follows. Extend the approach to consider moving targets and obstacles which may be intelligent to escape from the measurements of the sensors. Consider the movement of onboard sensors with respect to the robotic platforms since in this dissertation, all sensors are assumed to be fixed on the robotic platforms. Implement this method in a more general three-dimensional workspace in which assumption of prisms on the shape of targets and obstacles can be released. Apply this system to real outdoor applications, such as demining and cleaning.

Appendix A

Sensor Model Parameters

The sensor model parameters were chosen to simulate the environmental parameters' influence on the accuracy of the sensor measurements, while guaranteeing $\theta \in [0,\ 1]$. In the Bernoulli, Poisson, and binomial sensor models, $\eta_1 = \frac{0.1}{3}$, $\eta_2 = \frac{1}{3}$, $\eta_3 = \frac{2}{3}$, and $\eta_4 = 1$. For the Bernoulli sensor model,

$$A = \begin{pmatrix} 0.1 & 0.11 & 0.12 \\ 0.1 & 0.11 & 0.12 \\ -0.1 & 0.11 & -0.12 \\ -0.1 & -0.11 & -0.12 \end{pmatrix} \tag{A.1}$$

and for the Poisson and binomial sensor models:

$$A = \begin{pmatrix} 0.1 & 0.11 & -0.12 \\ 0.1 & 0.11 & -0.12 \\ 0.1 & 0.11 & -0.12 \\ 0.1 & 0.11 & -0.12 \end{pmatrix}. \tag{A.2}$$

Appendix B

Algorithm for Information Roadmap Milestone Distribution

Sampling Distribution Generation

given $\mathcal{W}, \mathcal{A}, \mathcal{S}, P(y_i, M_i)$, and $\mathcal{E}_i^0$ for $\forall i \in I_T$

discretize $\mathcal{C}$ into N_q configurations $C = \{\mathbf{q}^1, \ldots, \mathbf{q}^{N_\mathbf{q}}\}$

create an empty set of sample configurations, $Q = \emptyset$

initialize variables, and let $I(\mathbf{q}) = 0$, $U_Q = 0$ and $V_Q = 0$

for $\forall \mathbf{q}^i \in C$

  if $\mathbf{q}^i \in \mathcal{C}_{free}$

    $U(\mathbf{q}^i) = 1$ $I(\mathbf{q}^i) = 0$

    compute $V(\mathbf{q}^i)$ from (4.8)

  else

    $U(\mathbf{q}^i) = 0$ $V(\mathbf{q}^i) = 0$ $I(\mathbf{q}^i) = 1$

end

for $\forall \mathbf{q}_0^i \in C$

compute $f(\mathbf{b}' = \mathbf{q}^i | \mathbf{b} = \mathbf{q}_0^i)$ from (2.10)

end

end

$U_Q = \sum_{\mathbf{q}^i \in C} U(\mathbf{q}^i)$ $V_Q = \sum_{\mathbf{q}^i \in C} V(\mathbf{q}^i)$

for $\forall \mathbf{q}^i \in C$

$\pi_U(\mathbf{q}^i) = \frac{U(\mathbf{q}^i)}{U_Q}$

$\pi_V(\mathbf{q}^i) = \frac{V(\mathbf{q}^i)}{V_Q}$

$\pi_G(\mathbf{q}^i) = \sum_{\mathbf{q}_0^i \in \mathcal{C}} f(2\mathbf{q}^i - \mathbf{q}_0^i | \mathbf{q}_0^i) I(2\mathbf{q}^i - \mathbf{q}_0^i) I(\mathbf{q}_0^i)$

$\pi(\mathbf{q}^i) = v_2 \pi_V(\mathbf{q}^i) + v_1(1 - v_2)\pi_G(\mathbf{q}^i) + (1 - v_1)(1 - v_2)\pi_U(\mathbf{q}^i)$

end

Appendix C

Algorithm for Information Roadmap Path Search

Pseudocode of A^* search for τ^* in G

procedure $A^*(G, c_0, c_f, R, h)$

initialize T_s and OPEN as empty

install c_0 into T_s

OPEN := (c_0,OPEN); label c_0 as visited

while OPEN $\neq \emptyset$

find $c := \arg\min_c(f(c))$ in OPEN

if $c = c_f$

exit loop

end

for every neighbor c' of c in G

if c' is not visited

OPEN := (c',OPEN); mark c' as visited, and label it with a pointer to c

elseif $g(c') > g(c) + R(c, c')$

modify T_s by redirecting the pointer of c' toward c

remove c' from OPEN

OPEN := (c',OPEN)

end

end

end

if OPEN $\neq \emptyset$

return the minimum-cost path by tracing the pointers in T_s from c_f to c_0

else

return failure

end

Bibliography

[1] S. Ferrari and A. Vaghi, "Demining sensor modeling and feature-level fusion by bayesian networks," *IEEE Sensors*, vol. 6, pp. 471–483, 2006.

[2] C. Hofner and G. Schmidt, "Path planning and guidance techniques for an autonomous mobile cleaning robot," *Robotics and Autonomous Systems*, vol. 14, pp. 199–212, 1995.

[3] E. U. Acar, "Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods," *International Journal of Robotic Research*, vol. 22, pp. 7–8, 2003.

[4] C. Kreucher, K. Kastella, and A. Hero, "Multi-platform information-based sensor management," in *Proc. of the SPIE Defense Transformation and Network-Centric Systems Symposium*, vol. 5820, Orlando, FL, 2005, pp. 141–151.

[5] S. Ferrari and C. Cai, "Information-driven search strategies in the board game of CLUE®," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 39, no. 3, pp. 607–625, 2009.

[6] C. Cai and S. Ferrari, "Information-driven sensor path planning by approximate cell decomposition," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 39, no. 3, pp. 672–689, 2009.

[7] S. Ferrari, R. Fierro, B. Perteet, C. Cai, and K. Baumgartner, "A geometric optimization approach to detecting and intercepting dynamic targets using a mobile sensor network," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 292–320, 2009.

[8] R. Siegel, "Land mine detection," *IEEE Instrumentation and Measurement Magazine*, vol. 5, no. 4, pp. 22–28, 2002.

[9] T. Weigel, J. Gutmann, M. Dietl, A. Kleiner, and B. Nebel, "Cs freiburg: Coordinating robots for successful soccer playing," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 685–699, 2002.

[10] S. Ferrari, C. Cai, R. Fierro, and B. Perteet, "A multi-objective optimization approach to detecting and tracking dynamic targets in pursuit-evasion games," in *Proc. of the 2007 American Control Conference*, New York, NY, 2007, pp. 5316–5321.

[11] S. Ferrari, G. Zhang, and T. Wettergren, "Probabilistic track coverage in cooperative sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 99, pp. 1–13, 2010.

[12] D. Culler, D. Estrin, and M. Srivastava, "Overview of sensor networks," *Computer*, vol. 37, no. 8, pp. 41–49, 2004.

[13] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," in *Proc. 10^{th} International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, San Jose, CA, 2002, pp. 96–107.

[14] S. Russell and P. Norvig, *Artificial Intelligence A Modern Approach*. Upper Saddle River, NJ: Prentice Hall, 2003.

[15] G. D. Hager, *Task-Directed Sensor Fusion and Planning: A Computational Approach*. Boston: Kluwer Inc, 1990.

[16] G. D. Hager and M. Mintz, "Computational methods for task-directed sensor data fusion and sensor planning," *International Journal of Robotics Research*, vol. 10, pp. 285–313, 1991.

[17] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 293–313, 2002.

[18] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," *IEEE Signal Processing Magazine*, vol. 19, pp. 61–72, 2002.

[19] R. F. Stengel, *Optimal Control and Estimation*. Dover, 1994.

[20] W. Schmaedeke, "Information based sensor management," in *Proc. of SPIE Signal Processing, Sensor Fusion, and Target Recognition II*, vol. 1955, Orlando, FL, 1993, pp. 156–164.

[21] K. Kastella, "Discrimination gain to optimize detection and classification," *IEEE Transactions on Systems, Man and Cybernetics - Part A*, vol. 27, no. 1, pp. 112–116, 1997.

[22] C. Kreucher, K. Kastella, and A. Hero, "Sensor management using an active sensing approach," *Signal Processing*, vol. 85, pp. 607–624, 2005.

[23] K. J. Hintz, "A measure of the information gain attributable to cueing," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 21, pp. 237–244, 1991.

[24] ——, "Multi-process constrained estimation," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 21, pp. 434–442, 1991.

[25] G. Zhang, S. Ferrari, and M. Qian, "An information roadmap method for robotic sensor path planning," *Journal of Intelligent and Robotic Systems*, vol. 56, pp. 69–98, 2009.

[26] J. Denzler and C. M. Brown, "Information theoretic sensor data selection for active object recognition and state estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 145–157, 2002.

[27] C. Kreucher, A. Hero, and K. Kastella, "A comparison of task driven and information driven sensor management for target tracking," in *Proc. of the IEEE Conference on Decision and Control*, Seville, Spain, 2005, pp. 4004–4009.

[28] J. Principe, D. Xu, Q. Zhao, and J. Fisher, "Learning from examples with information theoretic criteria," *The Journal of VLSI Signal Processing*, vol. 26, no. 1-2, pp. 61–77, 2000.

[29] E. Gokcay and J. Principe, "Information theoretic clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158–171, 2002.

[30] D. Erdogmus and J. Principe, "Generalized information potential criterion for adaptive system training," *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1035–1044, 2002.

[31] D. Hall and S. McMullen, *Mathematical Techniques in Multisensor Data Fusion*. Norwood, MA: Artech House, 2004.

[32] K. Kastella and S. Musick, "Search for optimal sensor management," in *Proc. SPIE*, vol. 2759, 1996, pp. 318–321.

[33] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[34] C. Cai, S. Ferrari, and Q. Ming, "Bayesian network modeling of acoustic sensor measurements," in *Proc. IEEE Sensors*, Atlanta, GA, 2007, pp. 345–348.

[35] X. Liao and L. Carin, "Application of the theory of optimal experiments to adaptive electromagnetic-induction sensing of buried targets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 961–972, 2004.

[36] J. R. Spletzer and C. J. Taylor, "Dynamic sensor planning and control for optimally tracking target," *International Journal of Robotics Research*, vol. 22, no. 1, pp. 7–20, Jan. 2003.

[37] H. Choset, "Coverage for robotics: A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1-4, pp. 113–126, 2001.

[38] Z. Sun, D. Hsu, T. Jiang, H. Kurniawati, and J. H. Reif, "Narrow passage sampling for probabilistic roadmap planning," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1105–1115, 2005.

[39] N. M. Amato, O. B. Bayazit, L. K. Dale, C. Jones, and D. Vallejo, "Choosing good distance metrics and local planners for probabilistic roadmap methods," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 4, pp. 442–447, 2000.

[40] V. Boor, M. H. Overmars, and A. F. der Stappen, "The gaussian sampling strategy for probabilistic roadmap planners," *Proc. Conf. Robotics and Automation*, vol. 2, pp. 1018–1023, 1999.

[41] L. E. Kavraki, P. Svetska, J. C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration space," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[42] S. Ji, R. Parr, and L. Carin, "Nonmyopic multiaspect sensing with partially observable markov decision processes," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 2720–2730, 2007.

[43] N. Rao, S. Hareti, W. Shi, and S. Iyengar, "Robot navigation in unknown terrains: Introductory survey of non-heuristic algorithms," in *Technical Report ORNL/TM-12410*, Oak Ridge National Laboratory, Oak Ridge, TN, 1993.

[44] N. Rao, "Robot navigation in unknown generalized polygonal terrains using vision sensors," *IEEE Transactions on System, Man, and Cybernetics*, vol. 25, no. 6, pp. 947–962, 1995.

[45] A. Lazanas and J. C. Latombe, "Motion planning with uncertainty - a landmark approach," *Artificial Intelligence*, vol. 76, pp. 287–317, 1995.

[46] K. Song and C. C. Chang, "Reactive navigation in dynamic environment using a multisensor predictor," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 29, no. 6, pp. 870–880, 1999.

[47] M. Kazemi, M. Mehrandezh, and K. Gupta, "Sensor-based robot path planning using harmonic function-based probabilistic roadmaps," *Proc. ICAR '05, 12th International Conference on Advanced Robotics*, pp. 84–89, 2005.

[48] Z. Sun and J. Reif, "On robotic optimal path planning in polygonal regions with pseudo-eucledian metrics," *IEEE Transactions on Systems, Man, and Cybernetics - Part A*, vol. 37, no. 4, pp. 925–936, 2007.

[49] X.-C. Lai, S.-S. Ge, and A. Al-Mamun, "Hierarchical incremental path planning and situation-dependent optimized dynamic motion planning considering accelerations," *IEEE Transactions on Systems, Man, and Cybernetics- Part A*, vol. 37, no. 6, pp. 1541–1554, 2007.

[50] J. C. Latombe, *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[51] J. Ren and K. Mclsaac, "A hybird-systems approach to potential field navigation for a multi-robot team," in *Proc. of IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 3875–3880.

[52] S. Shimoda, Y. Kuroda, and K. Iagnemma, "Potential field navigation of high speed unmanned ground vehicles on uneven terrain," in *Proc. of IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 2839–2844.

[53] S. Ge and Y. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, 2000.

[54] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proc. of IEEE Conference on Robotics and Automation*, Sacramento, CA, 1991, pp. 1398–1404.

[55] E. Rimon and D. Kodischek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.

[56] A. Masoud, "Managing the dynamics of a harmonic potential field-guided robot in a cluttered environment," *IEEE Trans. on Industrial Electronics.*, vol. 56, no. 2, pp. 488–496, 2009.

[57] S. Waydo and R. Murray, "Vehicle motion planning using stream functions," in *Proc. of the 2003 IEEE International Conference on Robotics & Automation*, Taipei, Taiwan, 2003, pp. 2484–2491.

[58] J. Rosell and P. Iniguez, "Path planning using harmonic functions and probabilistic cell decomposition," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation - ICRA 2005*, Barcelona, Spain, 2005, pp. 1815–1820.

[59] G. Song and N. M. Amato, "A motion planning approach to folding: From paper craft to protein folding," in *In Proc. IEEE Int. Conf. Robot. Autom. (ICRA*, 2001, pp. 948–953.

[60] C. Scrapper and A. Takeuchi, "Using a priori data for prediction and pbject recognition in an autonomous mobile vehicle," *Proc. SPIE Unmanned Ground Vehicle Technology*, vol. 5083, pp. 414–418, 2003.

[61] E. Gelenbe and Y. Cao, "Autonomous search for mines," *European Journal of Operation Research*, vol. 108, pp. 319–333, 1998.

[62] A. O. Hero, B. Ma, O. Michael, and J. D. Gorman, "Applications of entropic spanning graphs," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 85–95, 2002.

[63] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* John Wiley and Sons, Inc., 1991.

[64] S. Perrin, E. Duflos, P. Vanheeghe, and A. Bibaut, "Multisensor fusion in the frame of evidence theory for landmines detection," *IEEE Transactions on Systems, Man, and Cybernetics - Part C*, vol. 34, no. 4, pp. 485–498, 2004.

[65] F. V. Jensen, *Bayesian Networks and Decision Graphs.* Springer-Verlag, 2001.

[66] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, pp. 197–243, 1995.

[67] K. Murphy, *How To Use Bayes Net Toolbox*. [Online]. Avaliable: http://www.ai.mit.edu/ murphyk/Software/BNT/bnt.html, 2004.

[68] S. Berchtold and B. Glavina, "A scalable optimizer for automatically generated manipulator motions," München, Germany, 1994, pp. 1796–1802,1994.

[69] G. Zhang and S. Ferrari, "An adaptive artificial potential function approach for geometric sensing," in *Proc. of IEEE International Conference on Decision and Control*, Shanghai, China, 2009, pp. 7903–7910.

[70] S. Ferrari, R. Fierro, and D. Tolic, "A geometric optimization approach to tracking maneuvering targets using a heterogeneous mobile sensor network," in *Proc. of the 2009 Conference on Decision and Control*, Cancun, MX, 2009.

[71] T. A. Wettergren, "Performance of search via track-before-detect for distributed sensor networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 314–325, January 2008.

[72] A. Poore and N. Rijavec, "A numerical study of some data association problems arising in multitarget tracking," *Computational Optimization and Applications*, vol. 3, no. 1, pp. 27–57, 1994.

[73] P. Morse and G. Kimball, *Methods of Operations Research*. Dover Publications, 2003.

[74] R. Hogg, J. McKean, and A. Craig, *Introduction to Mathematical Statistics*. Upper Saddle River, NJ: Prentice Hall, 2005.

[75] G. McLachlan, *Finite Mixture Models*. New York, NY: Wiley Interscience, 2000.

[76] T. Wettergren, R. Streit, and J. Short, "Tracking with distributed sets of proximity sensors using geometric invariants," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 4, pp. 1366–1374, 2004.

[77] C. A. McGilchrist, "Estimation in generalized mixed models," *Journal of the Royal Statistical Society*, vol. 56, no. 1, pp. 61–69, 1994.

[78] Mathworks, *Matlab*. [Online]. Avaliable: http://www.mathworks.com, 2004, functions: rand.

[79] M. Qian and S. Ferrari, "Probabilistic deployment for multiple sensor systems," in *Proc. of the 12th SPIE Symposium on Smart Structures and Materials: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, vol. 5765, San Diego, 2005, pp. 85–96.

[80] C. Cai and S. Ferrari, "Comparison of information-theoretic objective functions for decision support in sensor systems," in *Proc. American Control Conference*, New York, NY, 2007, pp. 63–133.

[81] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[82] N. J. Nilsson, *Principles of Artificial Intelligence*. Los Altos, CA: Morgan Kaufmann, 1980.

[83] R. E. Bellman and S. E. Dreyfus, *Applied Dynamic Programming*. Princeton, NJ: Princeton University Press, 1962.

[84] D. J. Daniels, "A review of GPR for landmine detection," *Sensing and Imaging: An International Journal*, vol. 7, no. 3, pp. 90–123, 2006.

[85] J. Paik, "Image processing-based mine detection techniques using multiple sensors: A review," *Subsurface Sensing Technologies and Applications, An International Journal*, vol. 3, pp. 203–252, 2002.

[86] J. MacDonald, *Alternatives for Landmine Detection*. Rand Publications, 2003.

[87] R. V. Dam, "Soil effects on thermal signatures of buried nonmetallic landmines," *Detection and Remediation Technologies for Mines and Minelike Targets VIII, Proc. of the SPIE*, vol. 5089, pp. 1210–1218, 2003.

[88] Explosive, Ordnance, Disposal, (EOD), and Technicians, *ORDATA Online*. Avaliable: http://maic.jmu.edu/ordata/mission.asp, 2006.

[89] R. Fierro and F. Lewis, "Control of nonholonomic mobile robot using neural networks," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 589–600, 1998.

[90] D. Cruz, J. McClintock, B. Perteet, O. Orqueda, Y. Cao, and R. Fierro, "Decentralized cooperative control - a multivehicle platform for research in networked embedded systems," *IEEE Control Systems Magazine*, vol. 27, pp. 58–78, 2007.

[91] A. Vaghi, *Sensor Management by a Graphical Model Approach.* Politecnico di Milano: Laurea Thesis, 2004.

[92] Mathworks, *Matlab.* [Online]. Avaliable: http://www.mathworks.com, 2004.

[93] E.-M. Arkin, S.-P. Fekete, and J.-S.-B. Mitchell, "Approximation algorithms for lawn mowing and milling," *Computational Geometry*, vol. 17, no. 1, pp. 25–50, 2000.

[94] M. Qian and S. Ferrari, "Probabilistic deployment for multiple sensor systems," *Proc. SPIE Symposium on Smart Structures and Materials*, 2005.

[95] S. M. LaValle, *Planning Algorithms.* Cambridge, U.K.: Cambridge University Press, 2006, available at http://planning.cs.uiuc.edu/.

[96] K. Pathak and S. Agrawal, "An integrated path-planning and control approach for nonholonomic unicycles using switched local potentials," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1201–1208, 2005.

[97] G. Casella and R. Berger, *Satistical Inference.* Duxbury Press, 2001.

[98] F. Lamiraux and J. P. Laumond, "On the expected complexity of random path planning," in *In IEEE Int. Conf. Robot. & Autom*, 1996, pp. 3306–3311.

[99] G. F. Lawler, *Introduction to Stochastic Processes.* Boca Raton, FL: Chapman & Hall/CRC, 2006.

[100] H. Khalil, *Nonlinear Systems.* Prentice Hall, 2002.

[101] W. Lu, G. Zhang, and S. Ferrari, "A randomized hybrid system approach to coordinated robotic sensor planning," in *IEEE Conference on Decision and Control (accepted)*, 2010.

Biography

Guoxian Zhang was born in June 1980 in Chengdu which is one of the major cities located in southwest of China. As most of other kids born in the 80s, he was the only child in his family and was taken good care of by his parents. When he was a little child, he showed curious on various appliance at home and began to tinker them. His parents always encouraged him to try all these things out and helped him build the interest in engineering. Therefore, when he entered Tsinghua University, he chose Mechanical Engineering as his Major. This was his first time to be away from his parents and his city to pursue his dream. During four years of studies, he not only focused on his course studies and ranked third over more than one hundred students in his department, but also took part in various competitions and activities held by different departments, such as mechanical engineering, electric engineering and civil engineering, and won numbers of prizes.

Guoxian's research began during his undergraduate program at Tsinghua University, Beijing, China when he worked on a team project of developing a humanoid robot. He continued his master study on the macro/micro manipulator project. He designed and manipulated a micro robot including its mechanism, circuit, and control algorithm. This micro robot was utilized by him in a macro/micro manipulator system to increase the trajectory accuracy of an industry robot. In his PhD study, he worked on an interdisciplinary project which includes area in mechanical engineering, computer science and statistics. Therefore, he covered a wide research subjects including sensor path planing, robotics, Bayesian data analysis, computational in-

telligence, and sensor fusion. His current research interests lie in: multiple agents systems, machine learning, and modern data analysis.